

FIG. 1

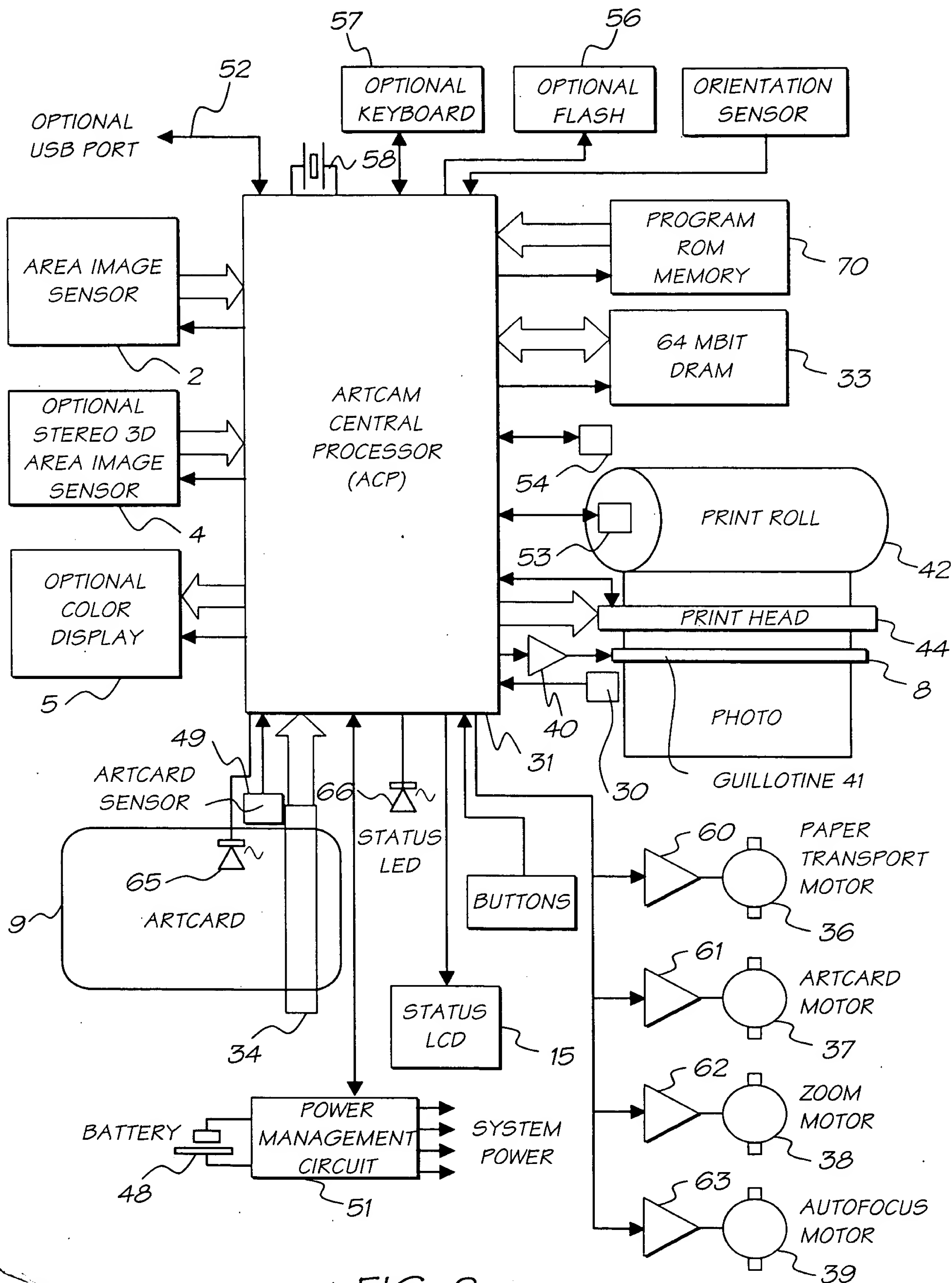


FIG. 2

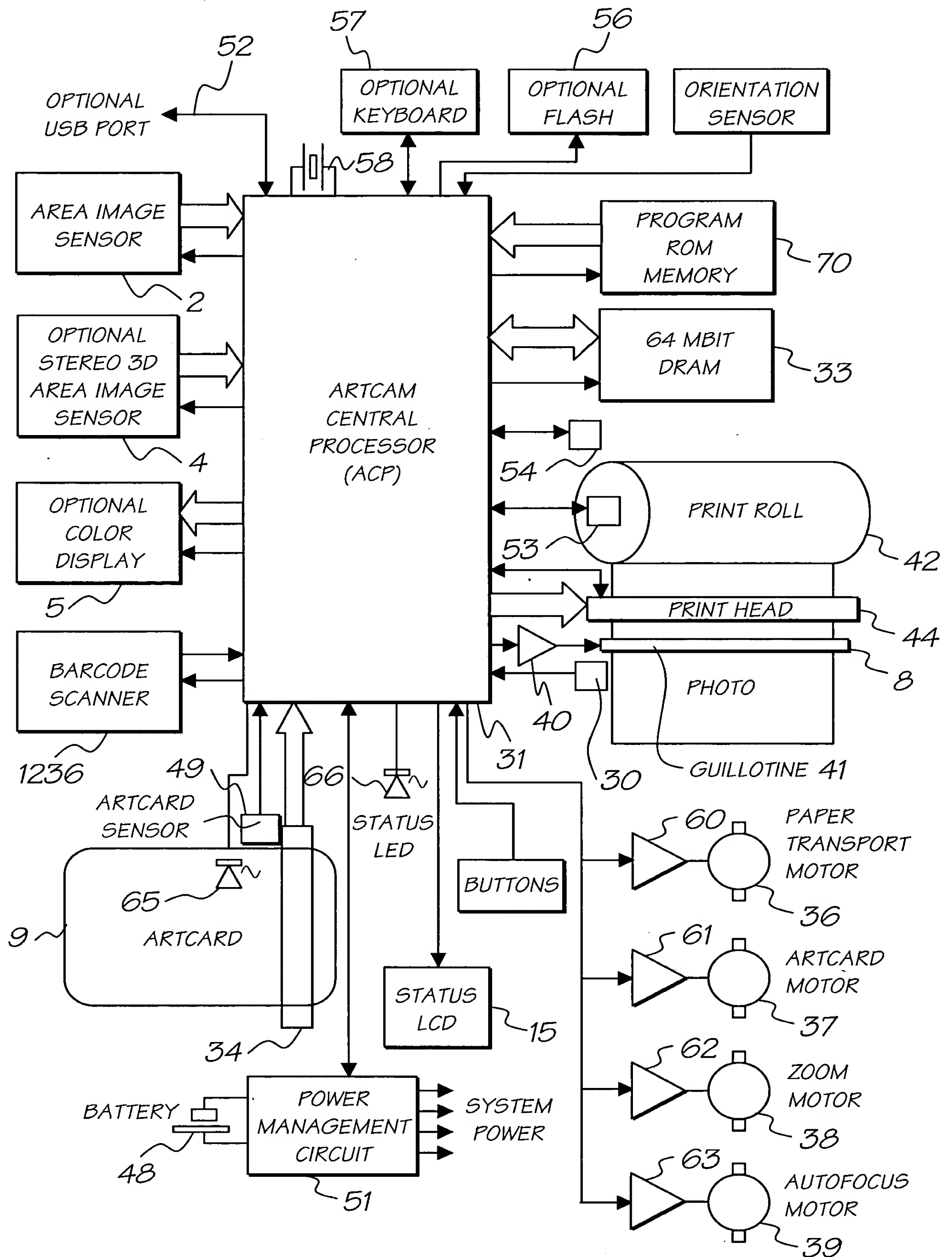


FIG. 2A

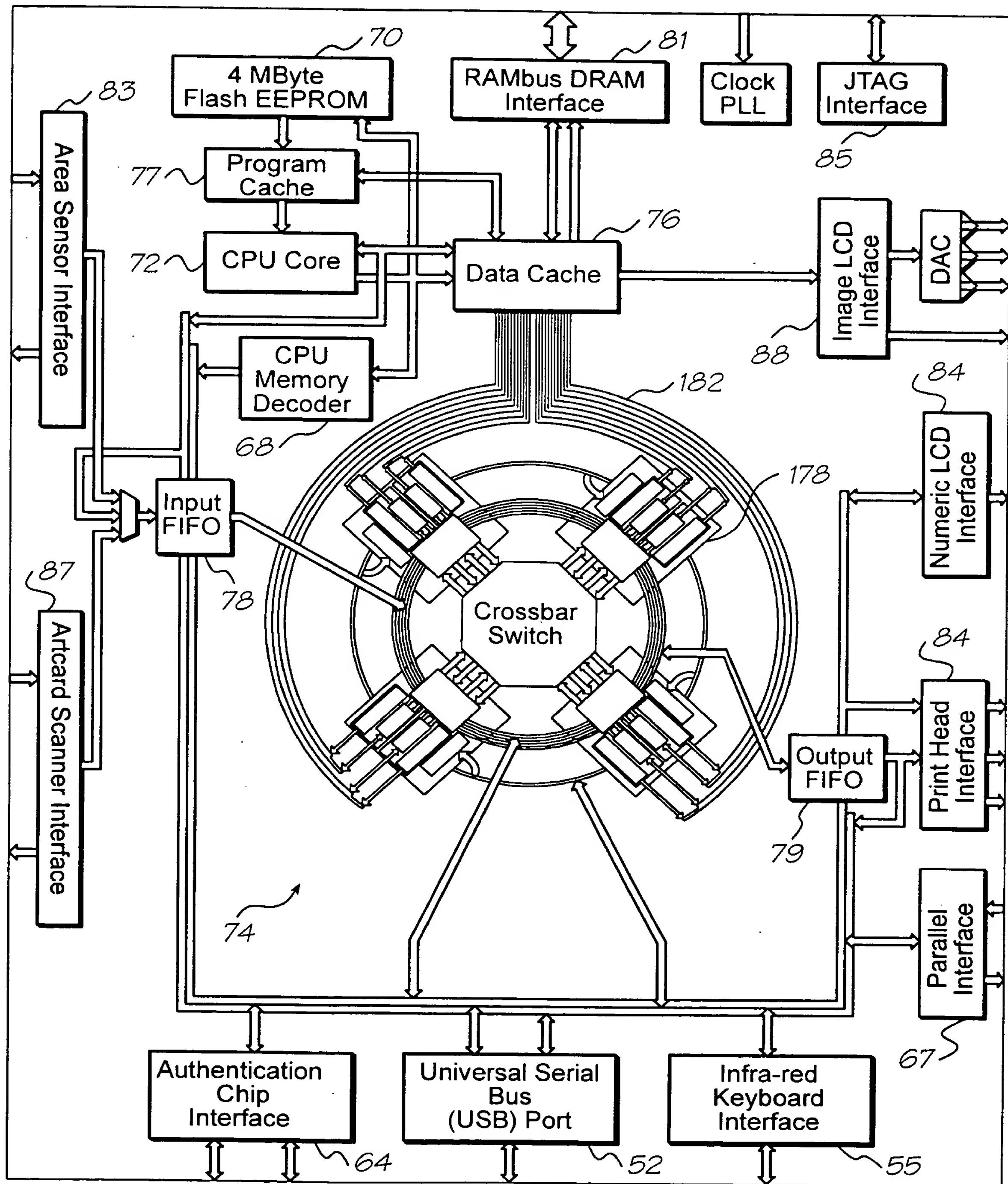


FIG. 3

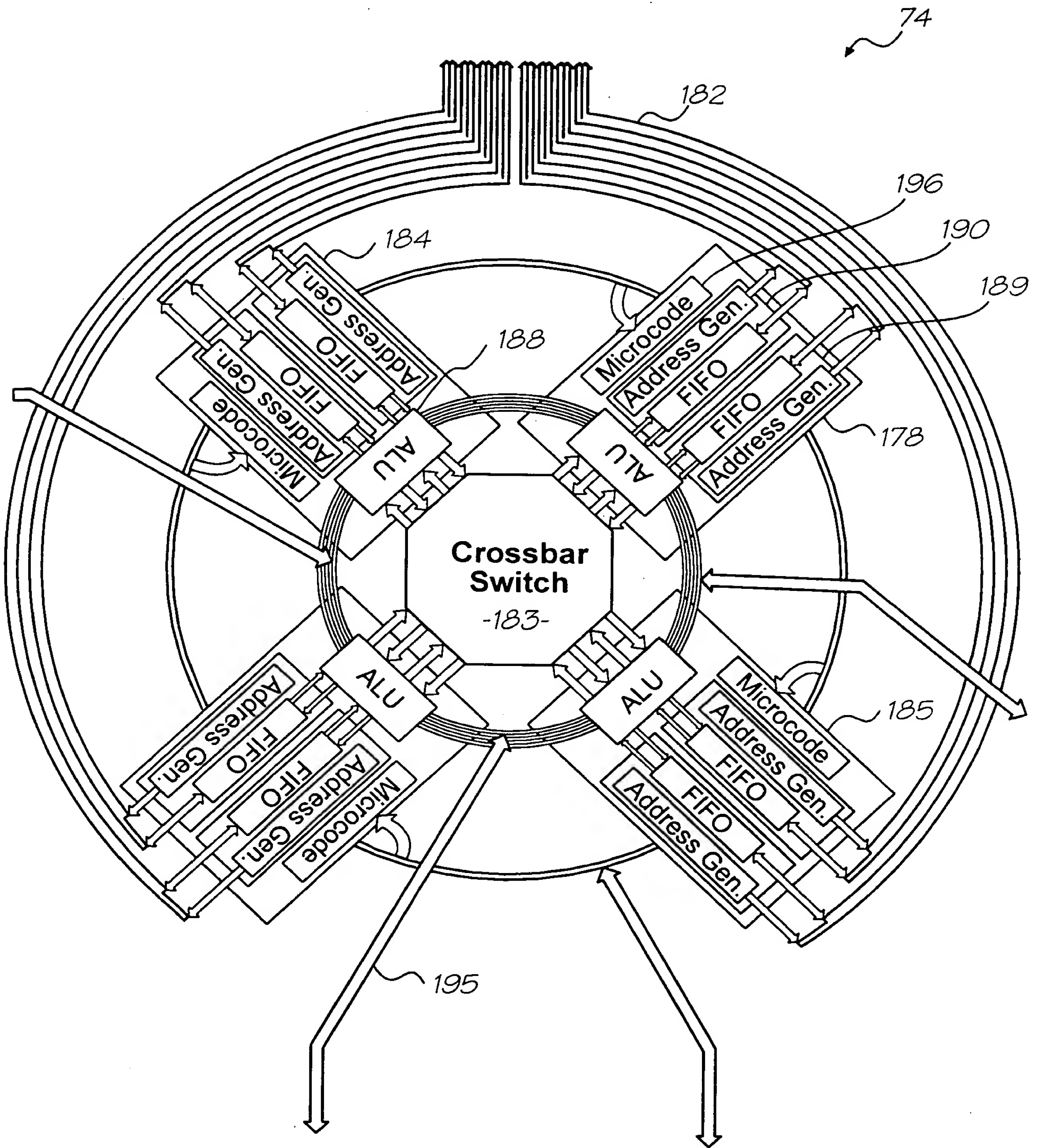


FIG. 3(a)

188

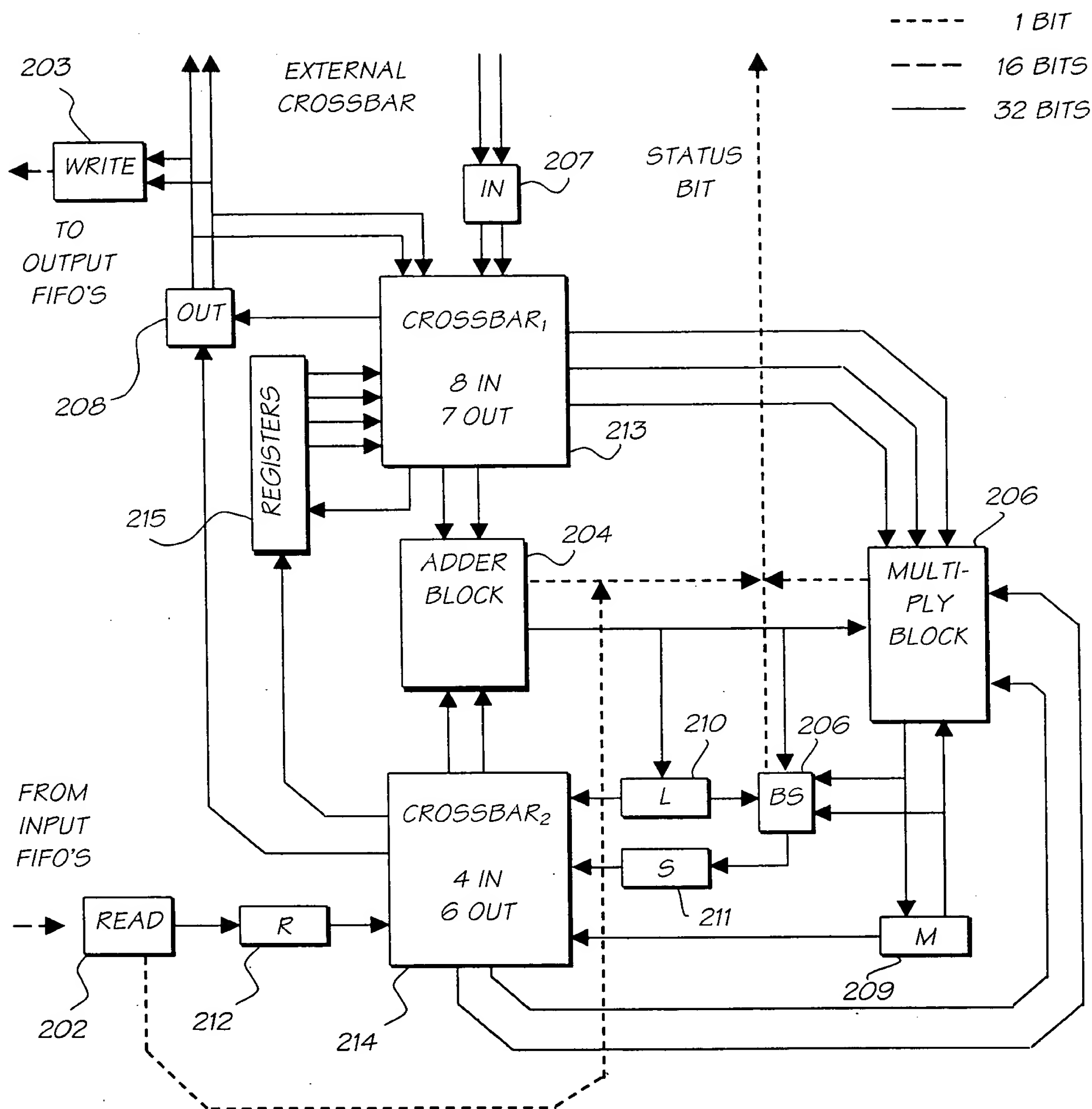


FIG. 5

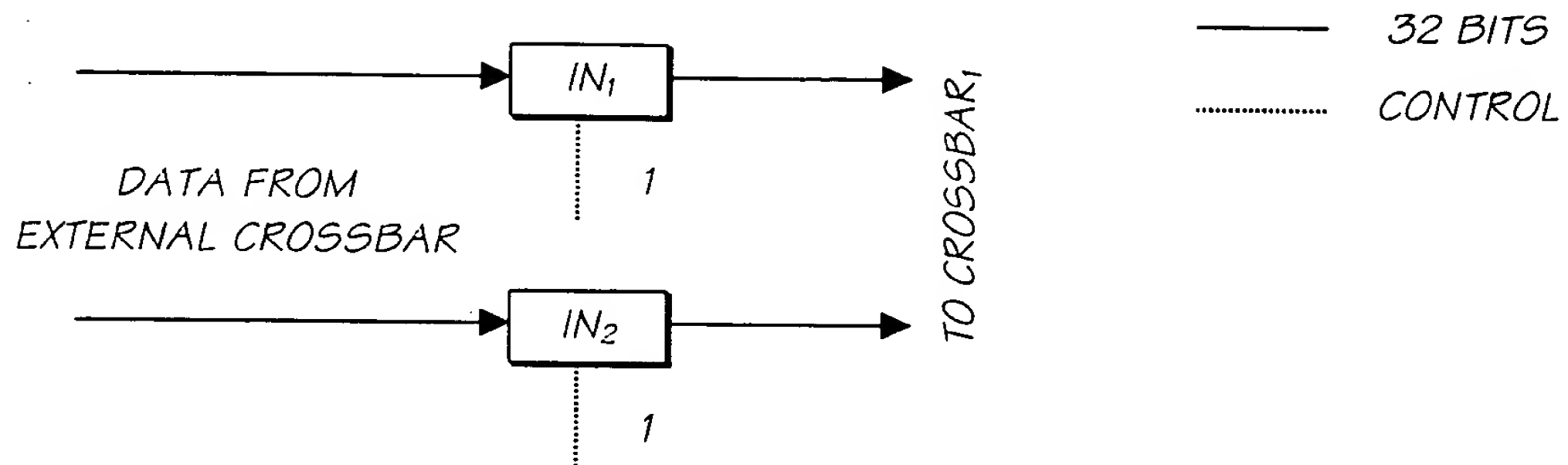


FIG. 6

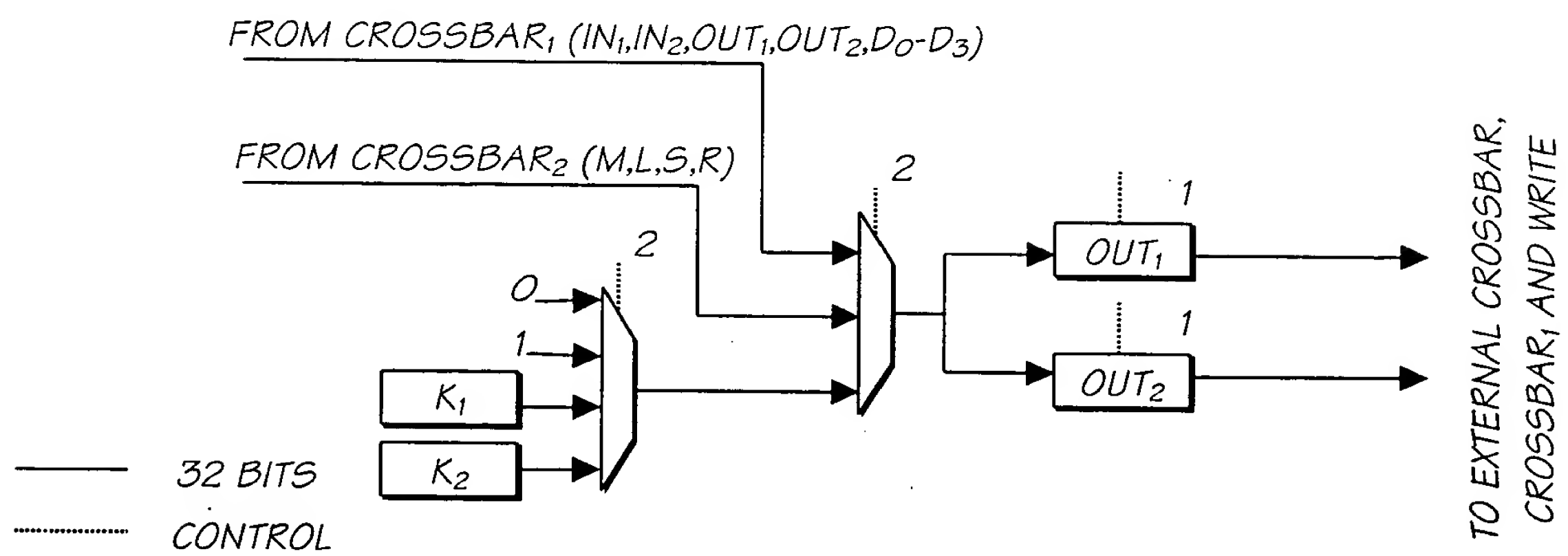


FIG. 7

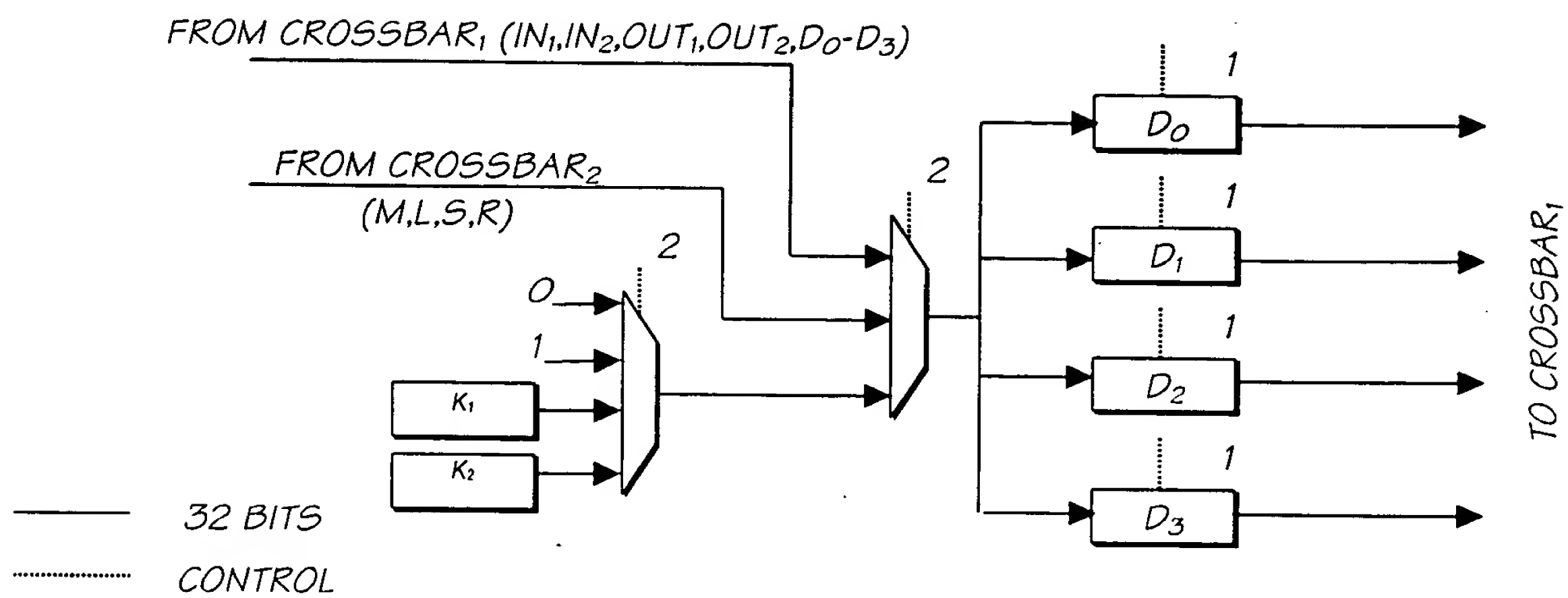


FIG. 8

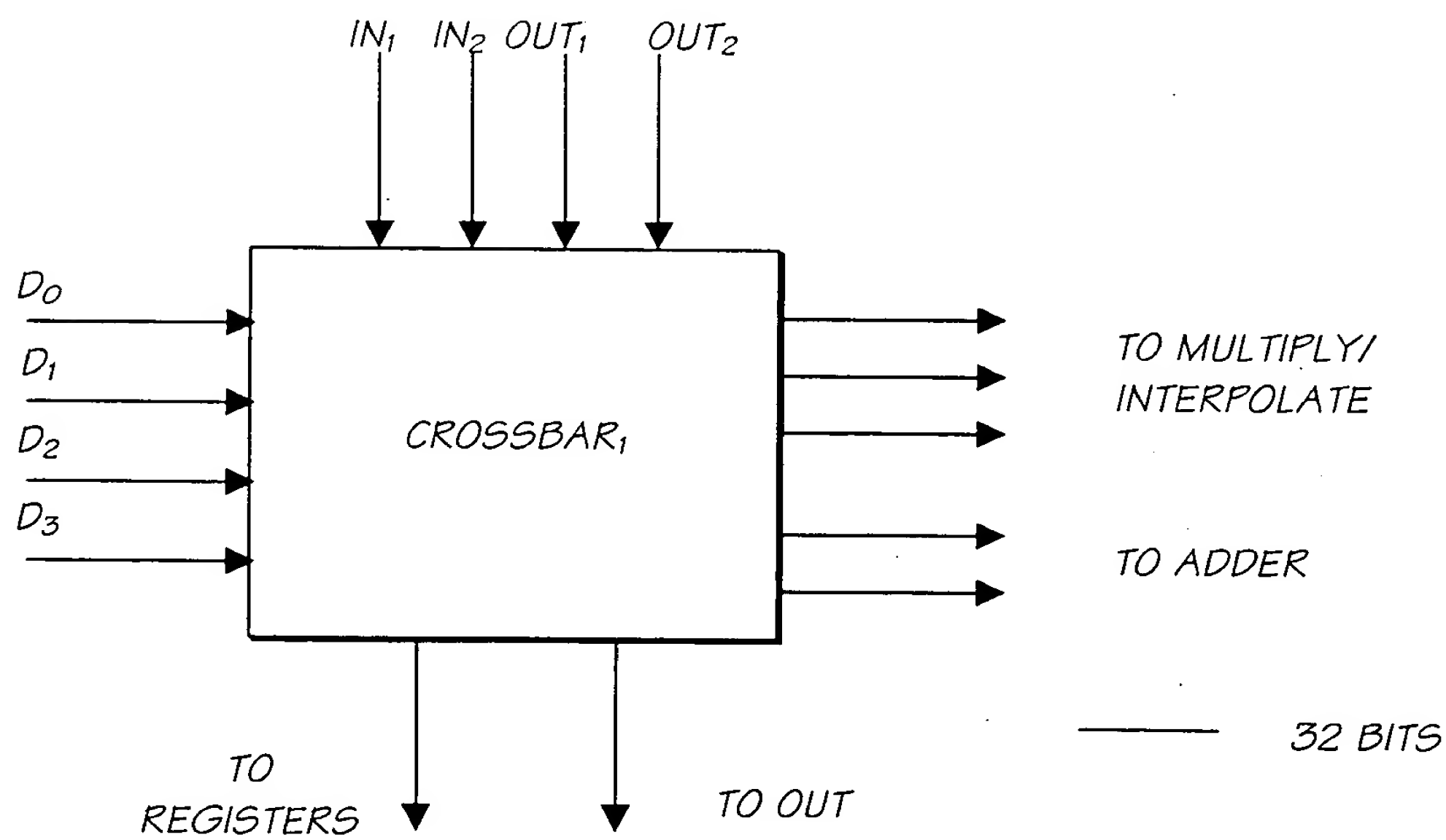


FIG. 9

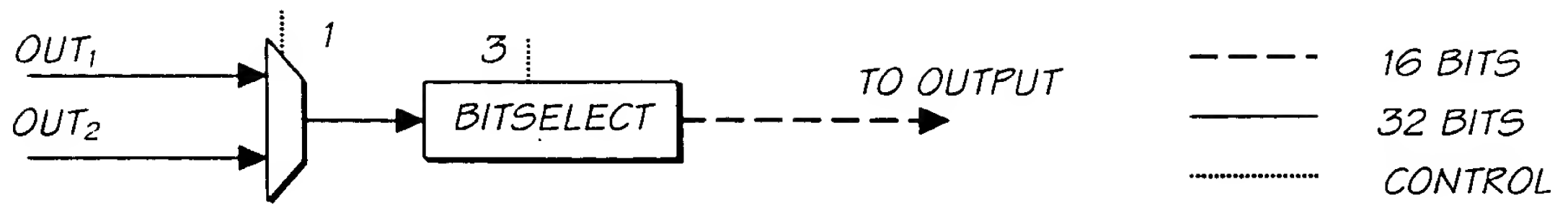


FIG. 12

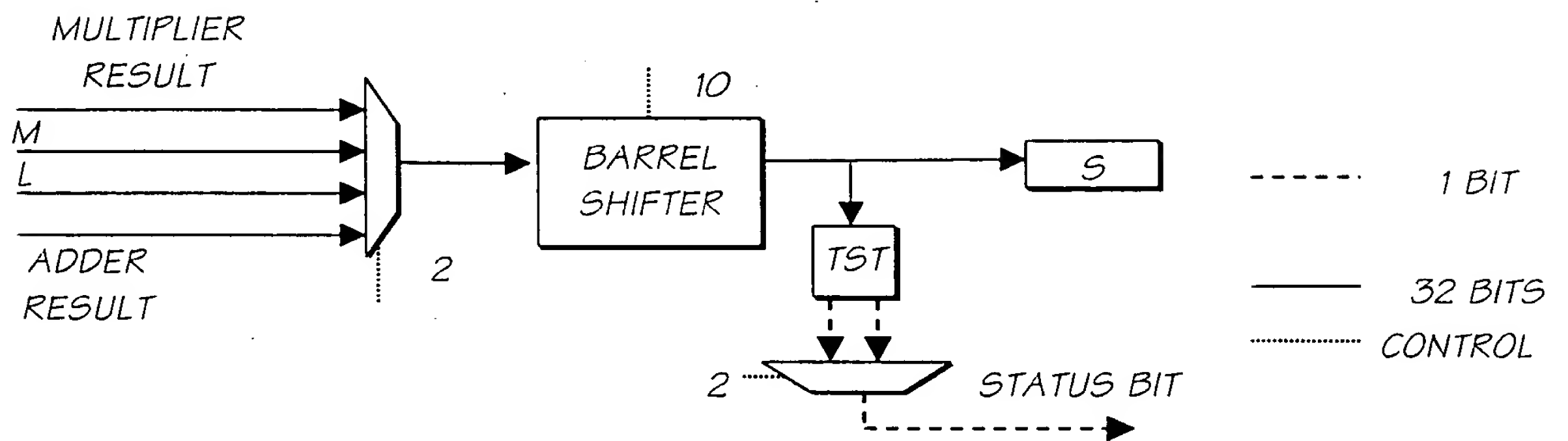


FIG. 13

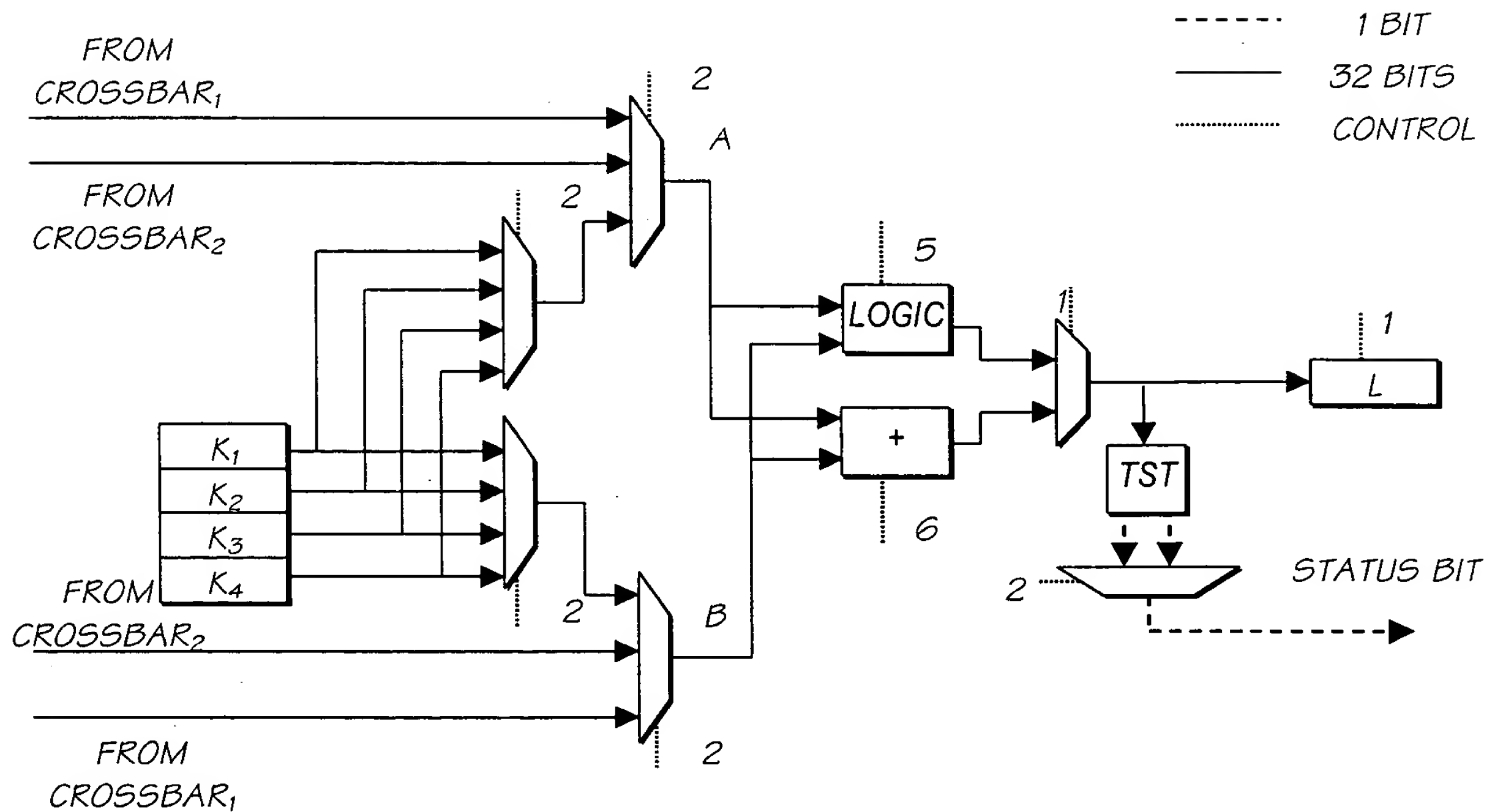


FIG. 14

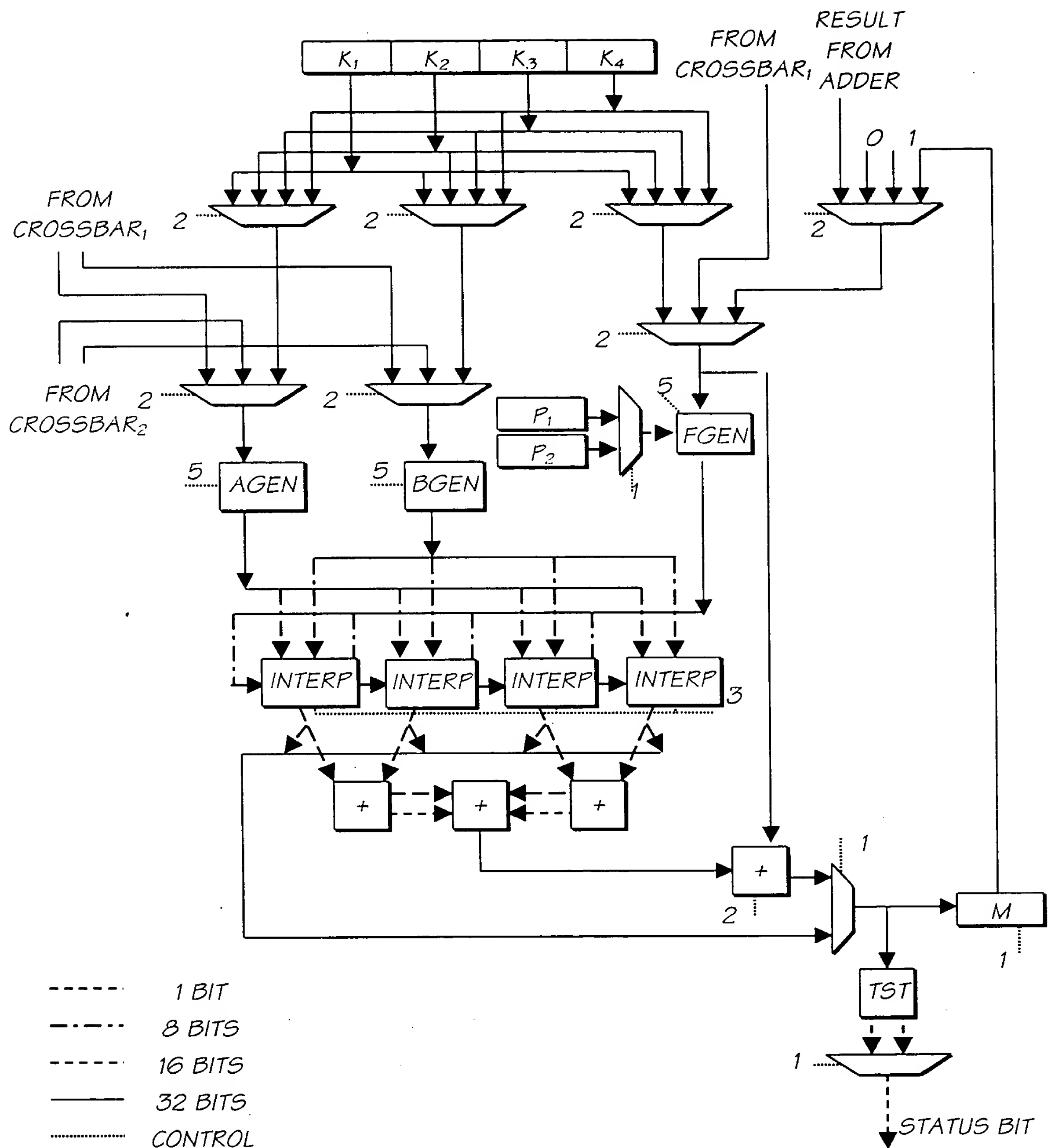


FIG. 15

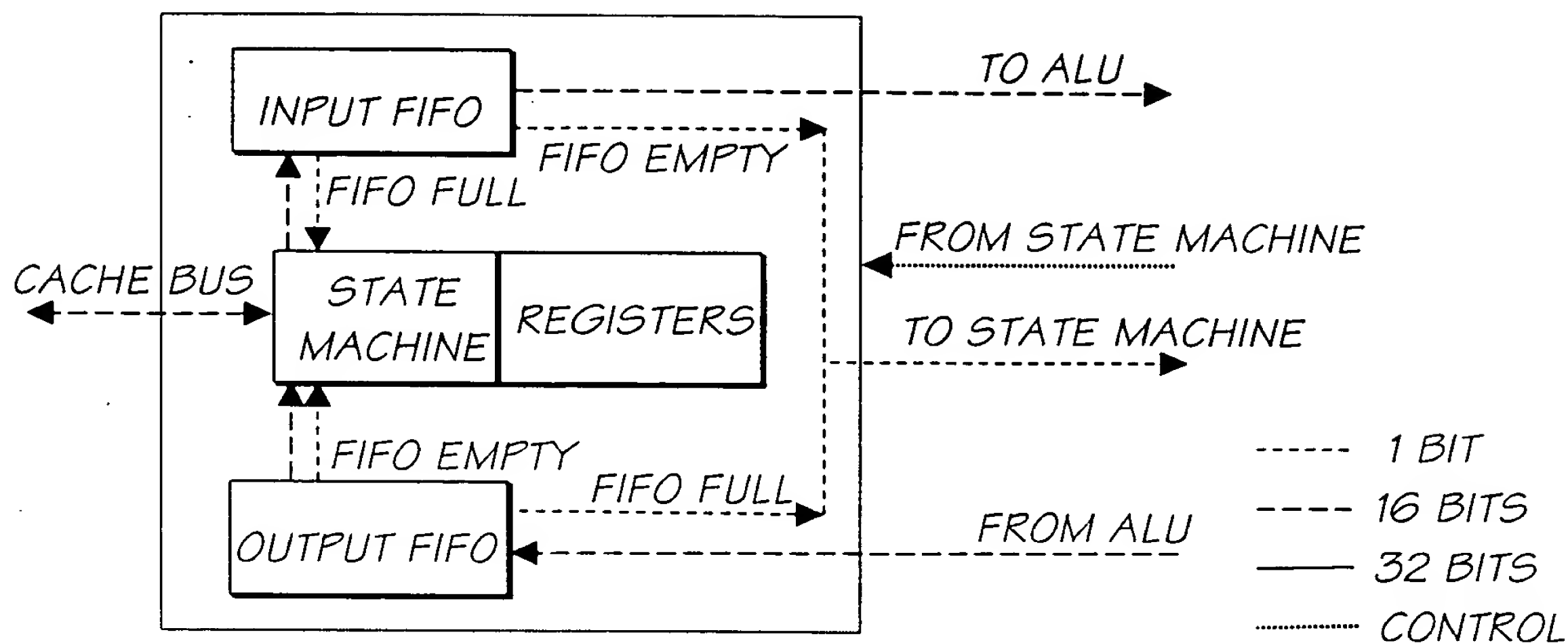


FIG. 16

ORDER OF PIXELS PRESENTED BY A SEQUENTIAL READ ITERATOR
ON A 4 X 2 IMAGE WITH PADDING.

0	1	2	3	
4	5	6	7	

FIG. 17

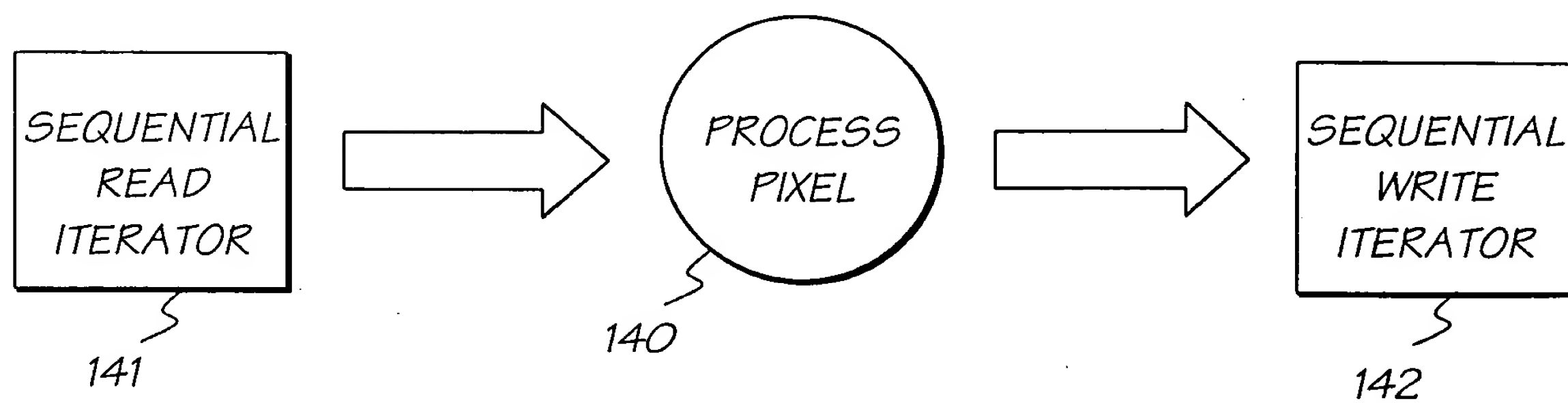
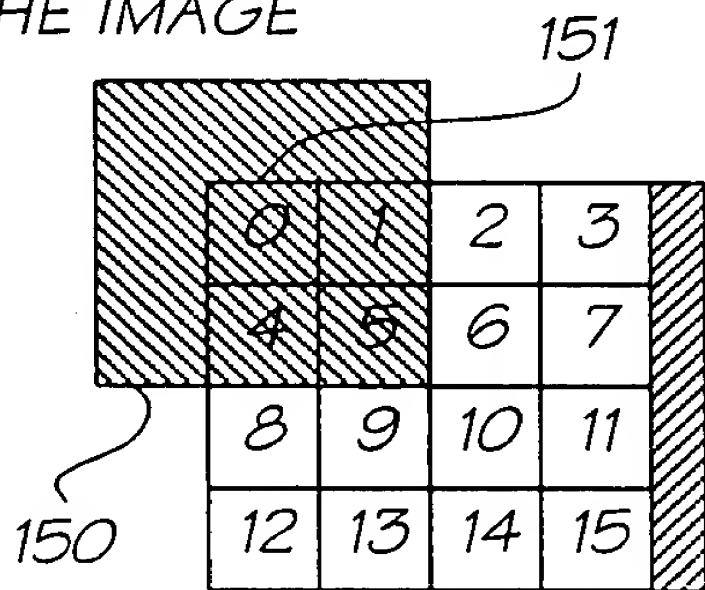


FIG. 18

A 3x3 BOX VIEW TRAVERSES THE PIXELS IN ORDER: 0, 1, 2, 3, 4, 5, 6, 7, 8
ETC, PLACING A 3x3 BOX CENTERED OVER EACH PIXEL...

3x3 BOX VIEW OF FIRST
PIXEL IN IMAGE = 9 PIXELS,
5 OF WHICH ARE OUTSIDE
THE IMAGE

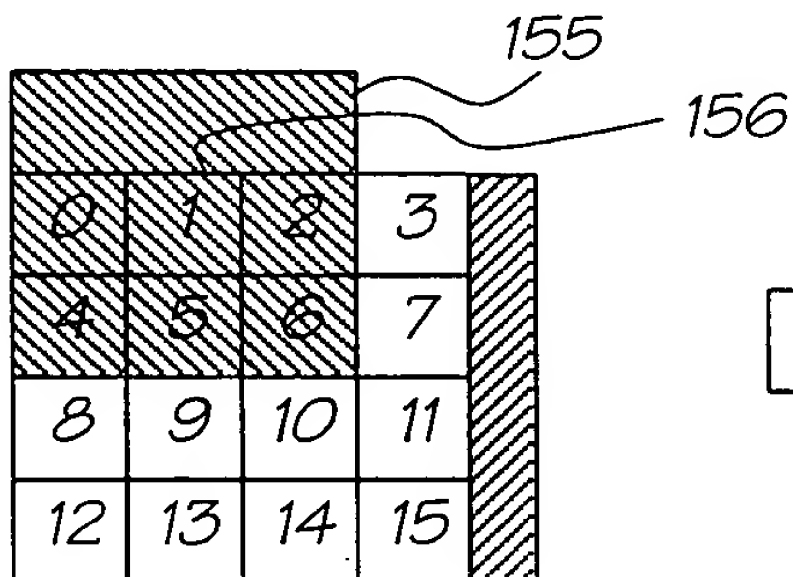


FIRST 9 PIXELS FROM THE BOX
READ ITERATOR:

IF DUPLICATION OF EDGE PIXELS IS
ON: 0, 0, 0, 0, 0, 1, 4, 4, 5

IF DUPLICATION OF EDGE PIXELS IS
OFF: V, V, V, V, 0, 1, V, 4, 5
WHERE V IS CONSTANTPIXEL
REGISTER VALUE REPRESENTING
"OUTSIDE THE IMAGE"

3x3 BOX VIEW OF
SECOND PIXEL IN IMAGE
= 9 PIXELS,
3 OF WHICH ARE
OUTSIDE THE IMAGE



SECOND 9 PIXELS FROM THE BOX
READ ITERATOR:

IF DUPLICATION OF EDGE PIXELS
IS ON: 0, 1, 2, 0, 1, 2, 4, 5, 6

IF DUPLICATION OF EDGE PIXELS
IS OFF: V, V, V, 0, 1, 2, 4, 5, 6
WHERE V IS CONSTANTPIXEL
REGISTER VALUE REPRESENTING
"OUTSIDE THE IMAGE"

FIG. 19

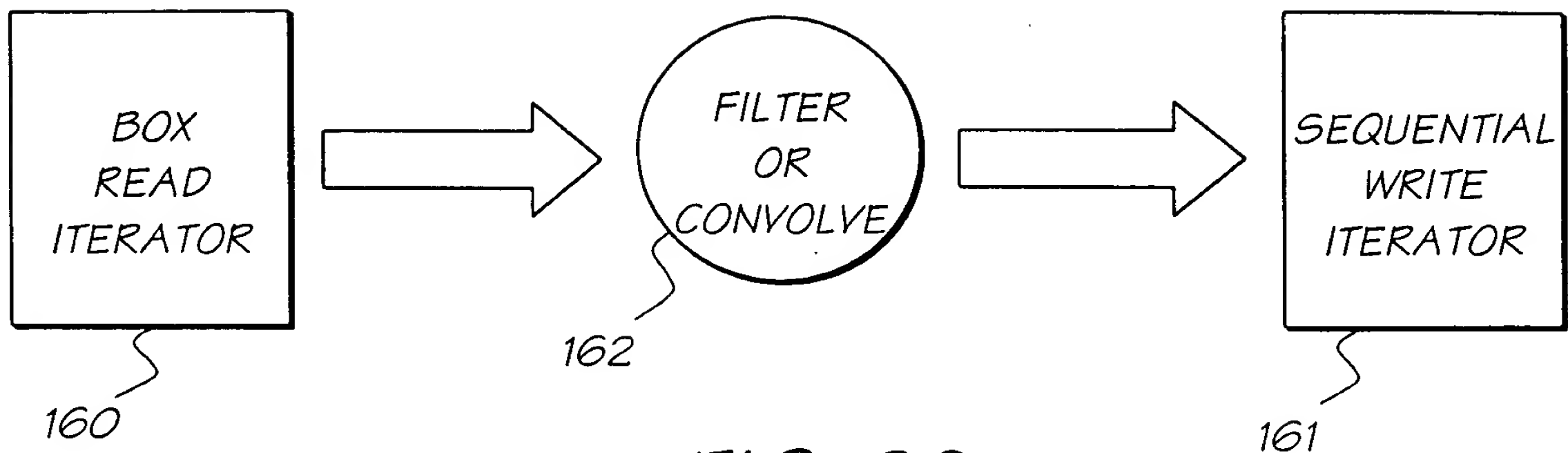
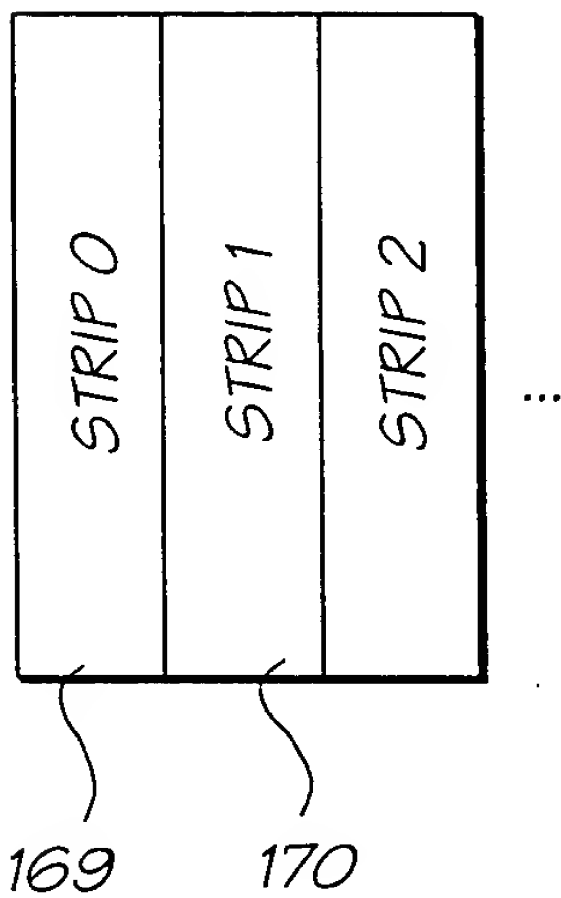
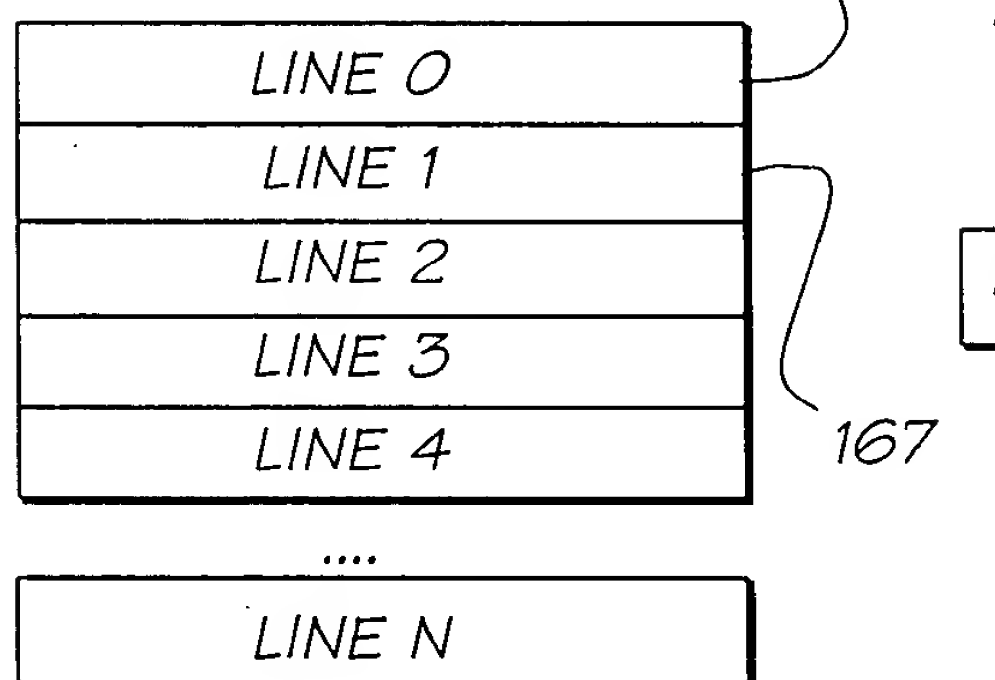


FIG. 20

IMAGE BROKEN INTO
VERTICAL STRIPS,
EACH STRIP IS 32
PIXELS ACROSS



LINES ARE ACCESSED
LINE 0 TO LINE N
WITHIN A SINGLE STRIP.



PIXELS ARE ACCESSED
PIXEL 0 - PIXEL 31
WITHIN A SINGLE LINE

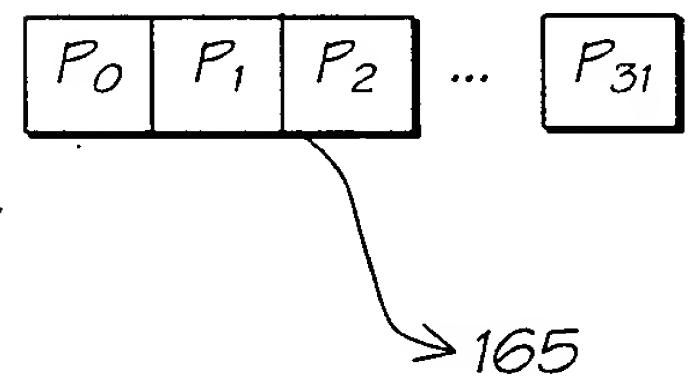


FIG. 21

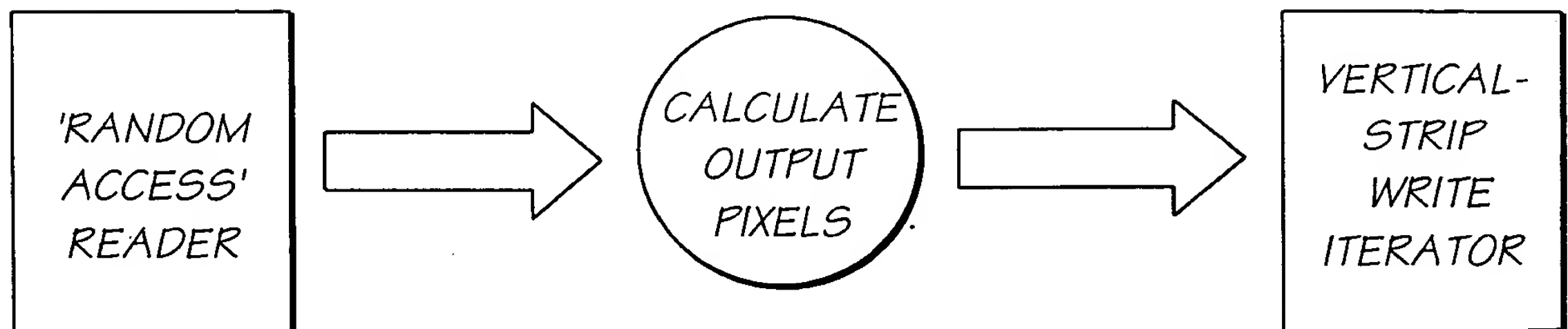


FIG. 22

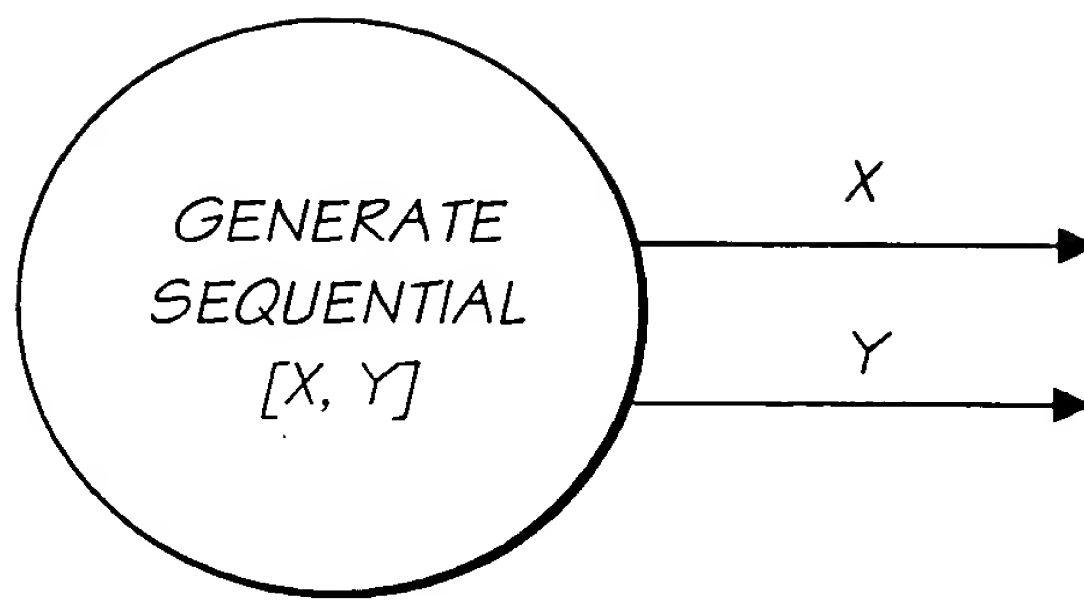


FIG. 23

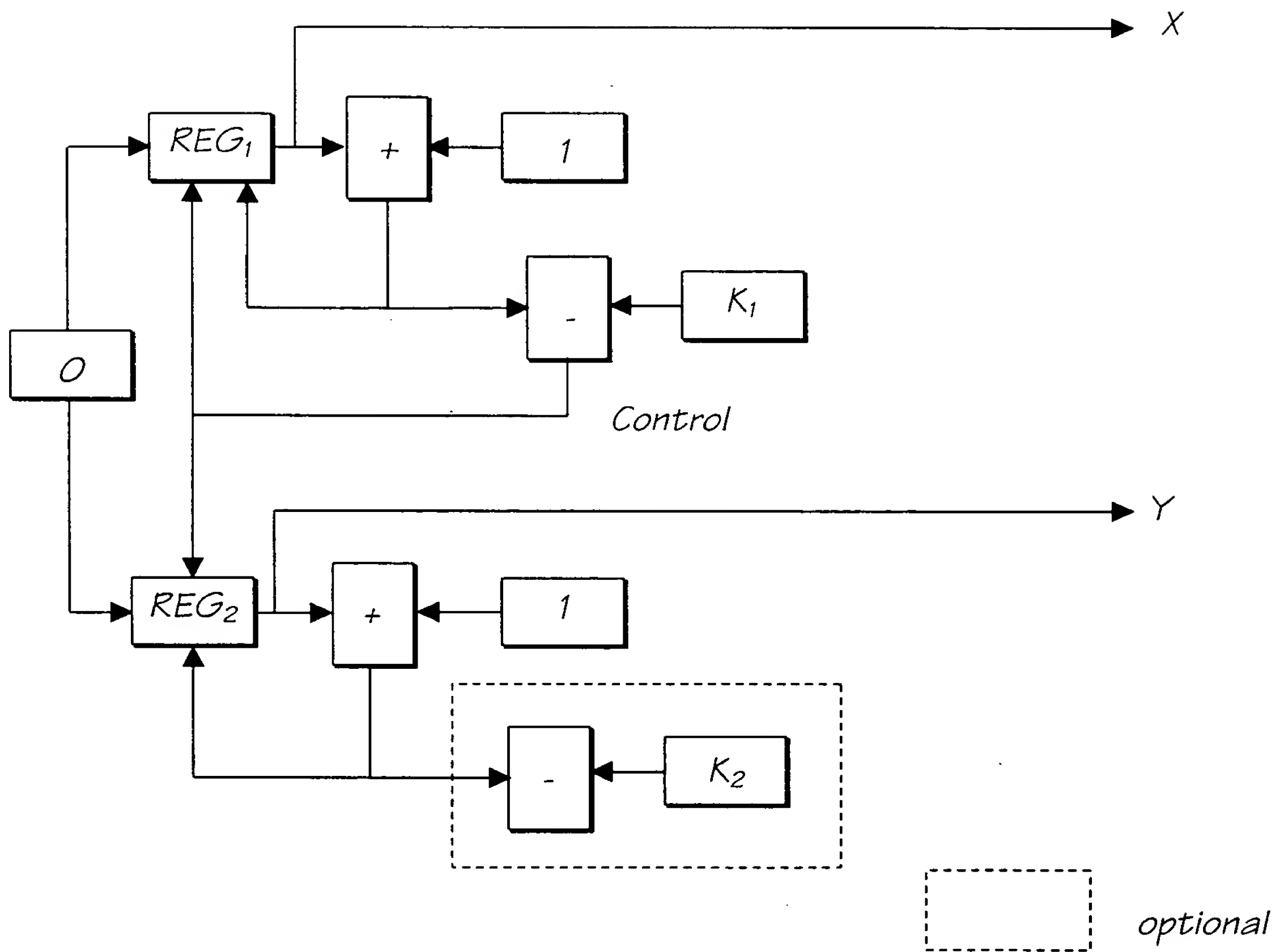


FIG. 24

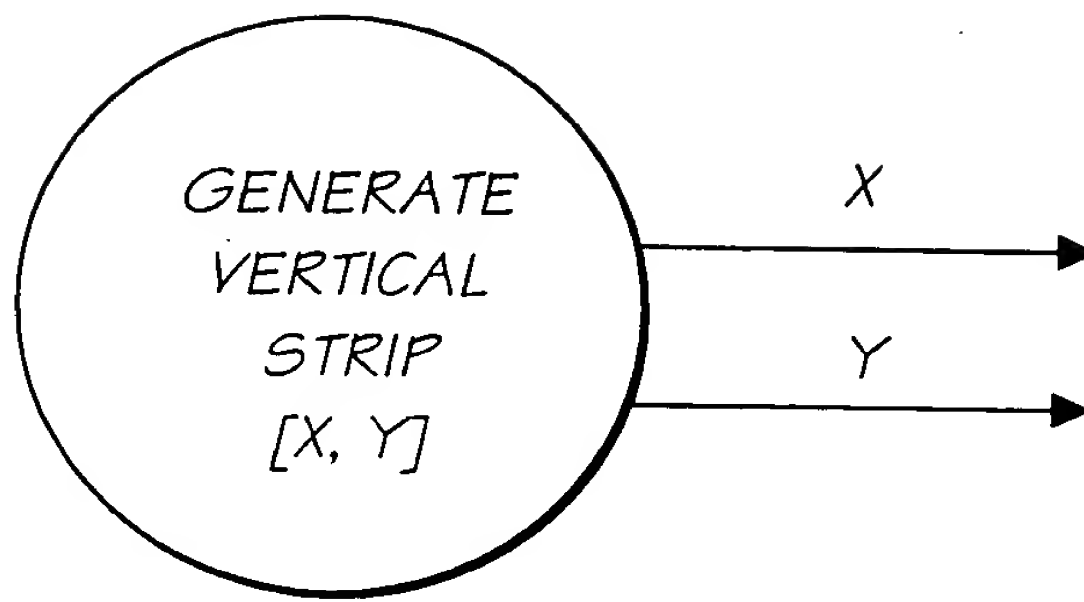


FIG. 25

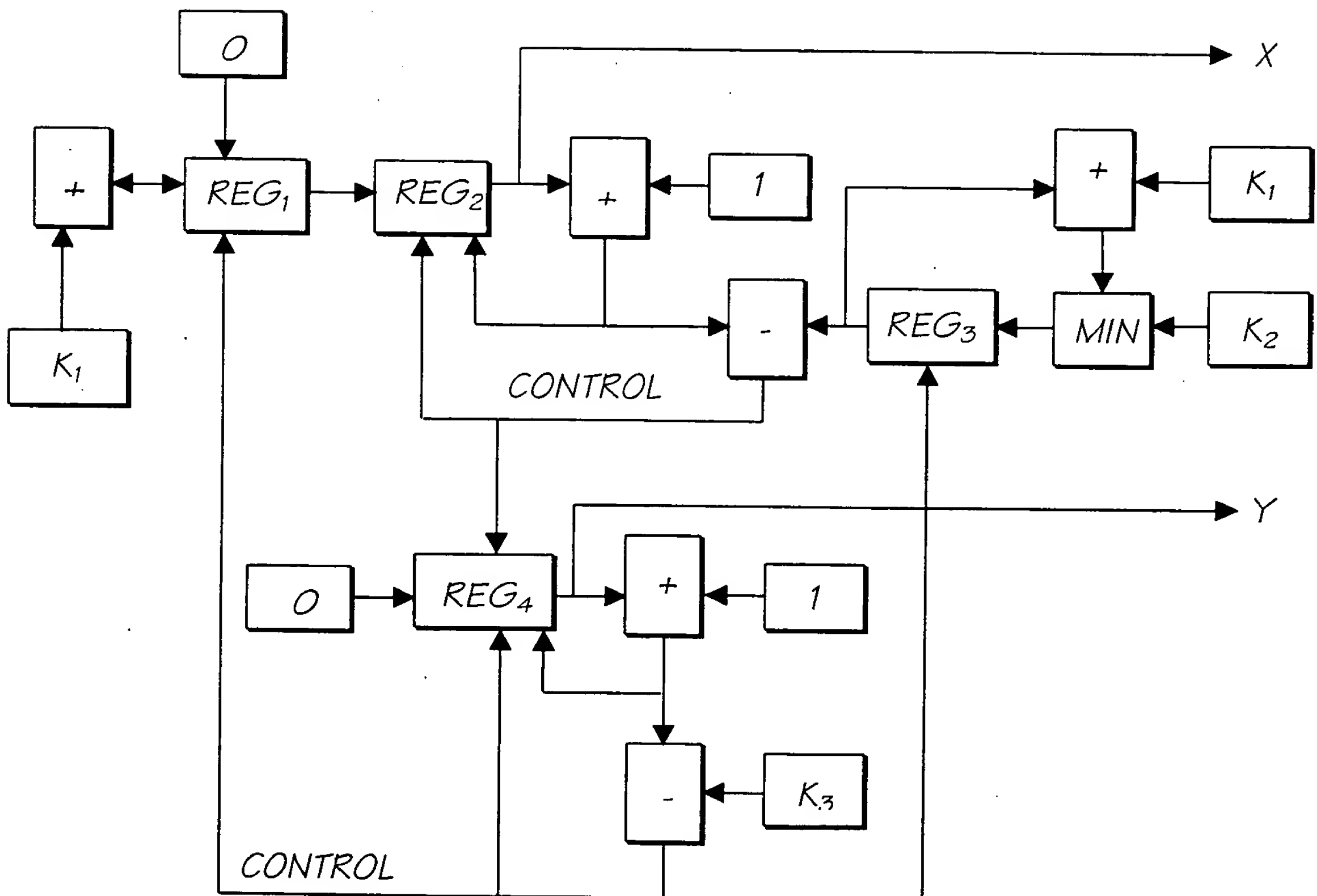
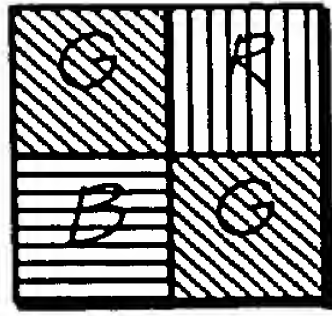


FIG. 26



2X2 PIXEL BLOCK FROM SENSOR

FIG. 27

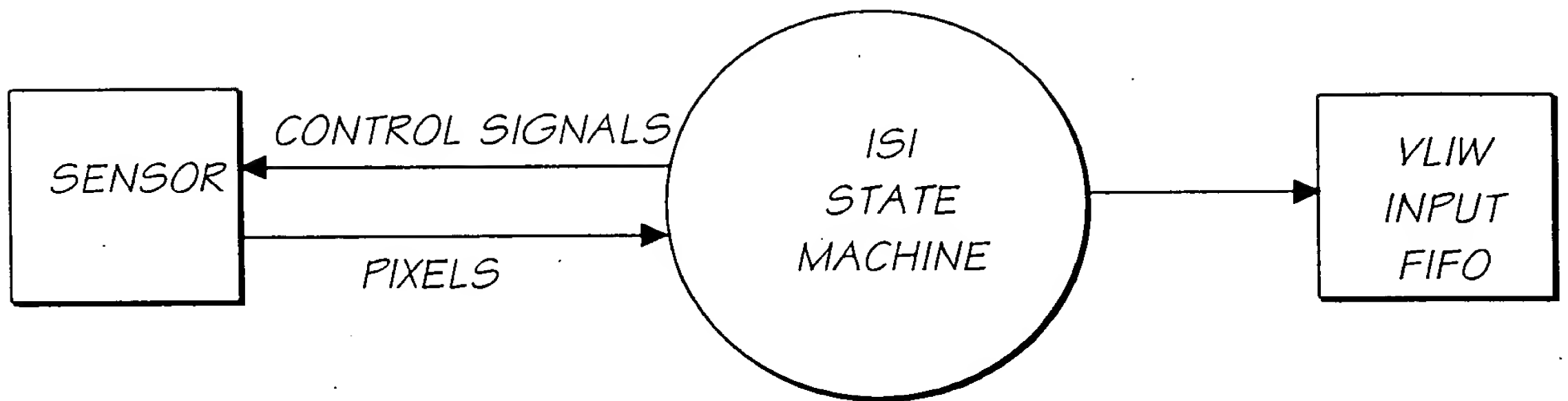


FIG. 28

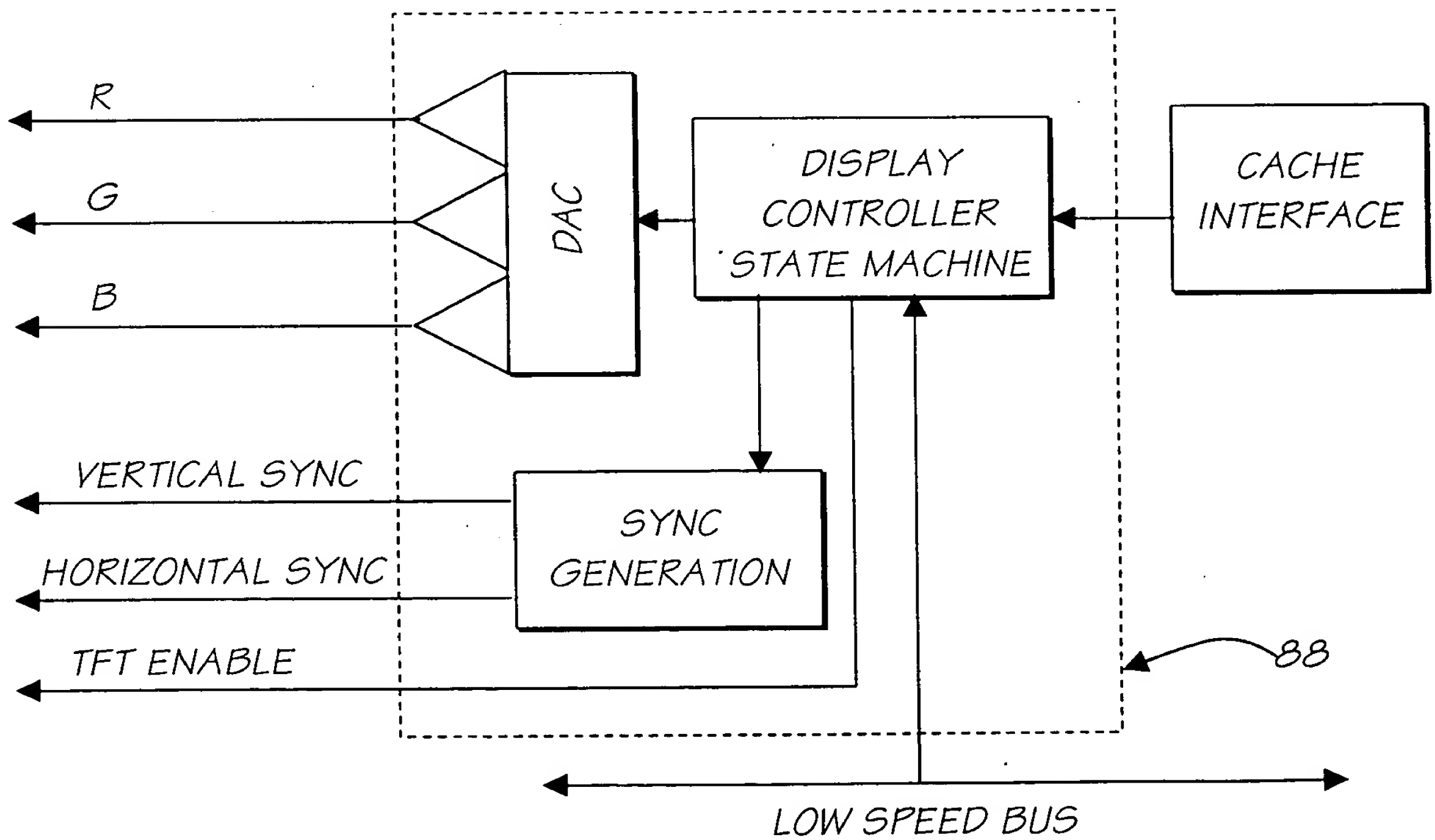
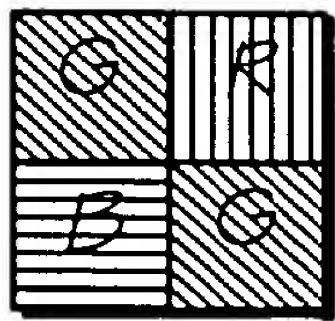


FIG. 29



2X2 PIXEL BLOCK FROM CCD

FIG. 30

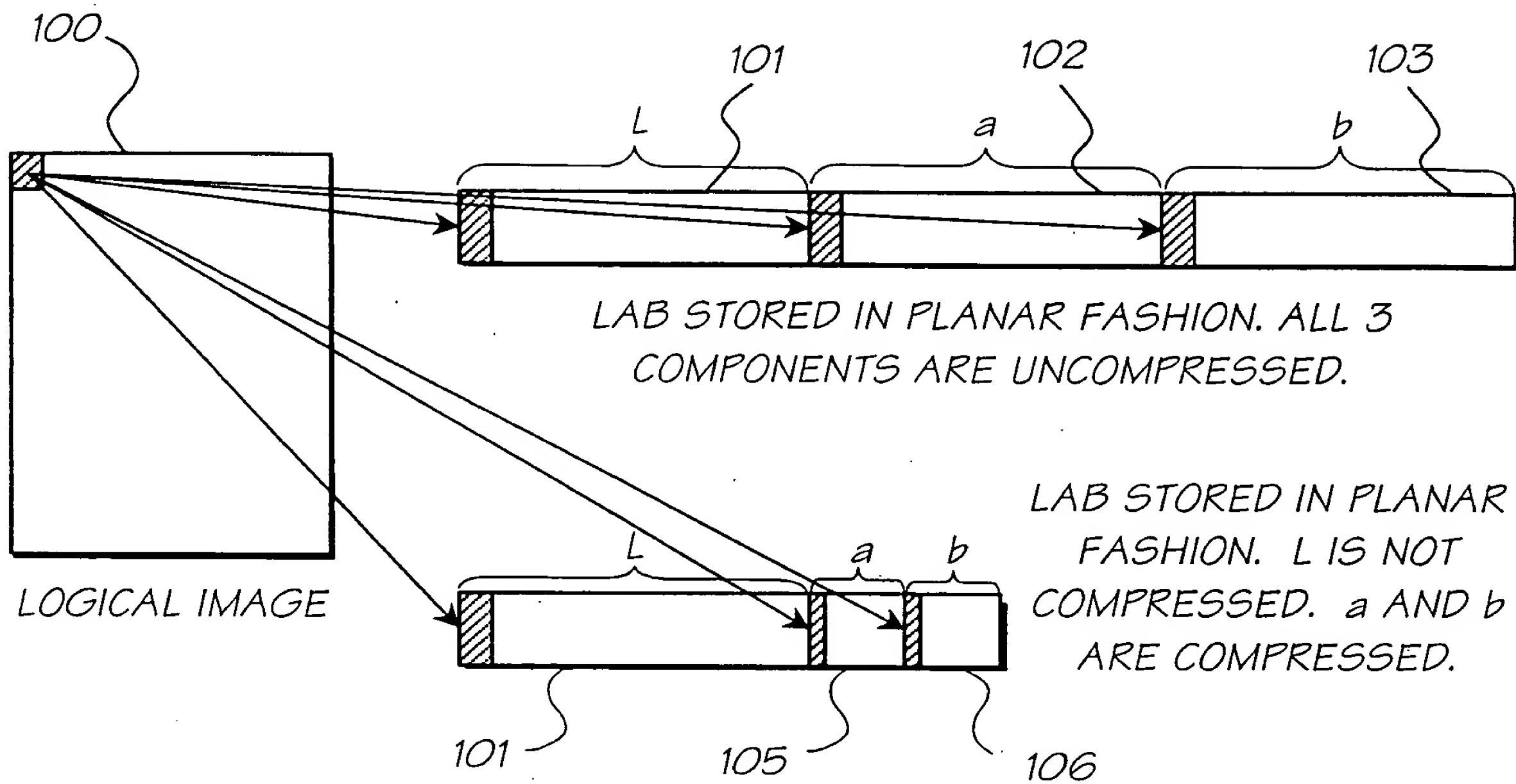


FIG. 31

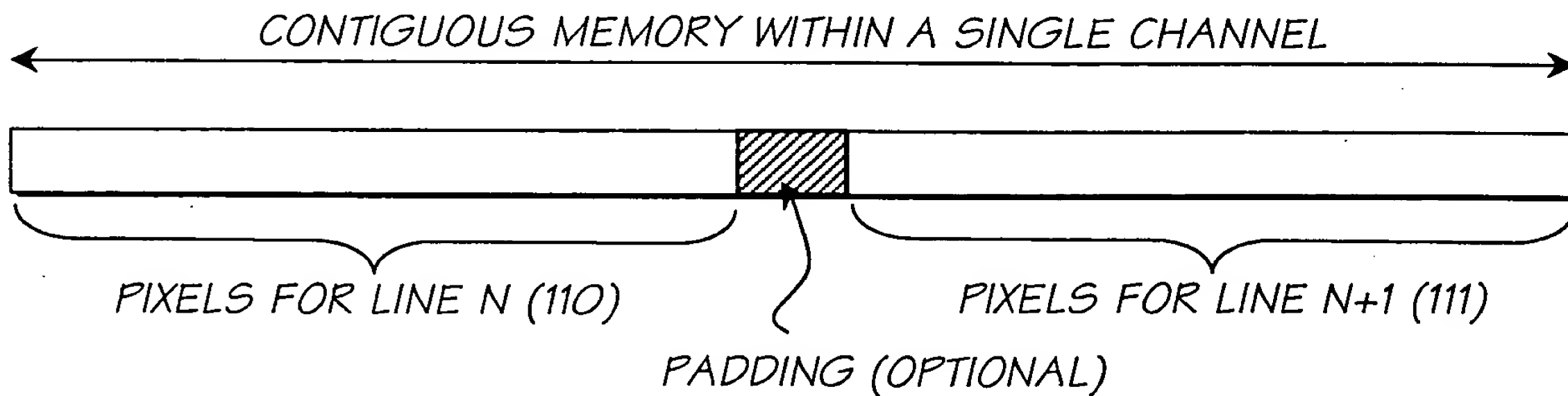


FIG. 32

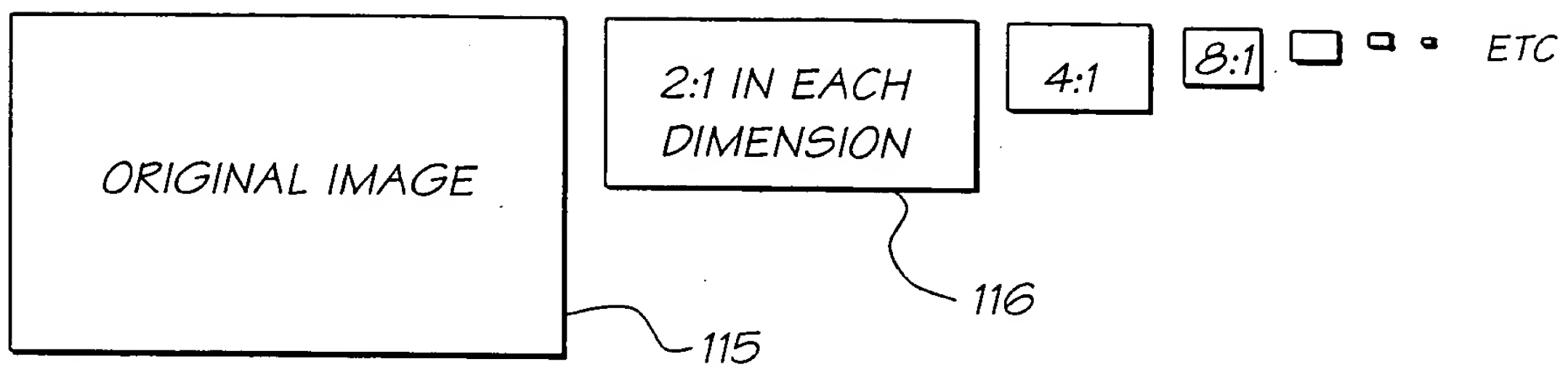


FIG. 33

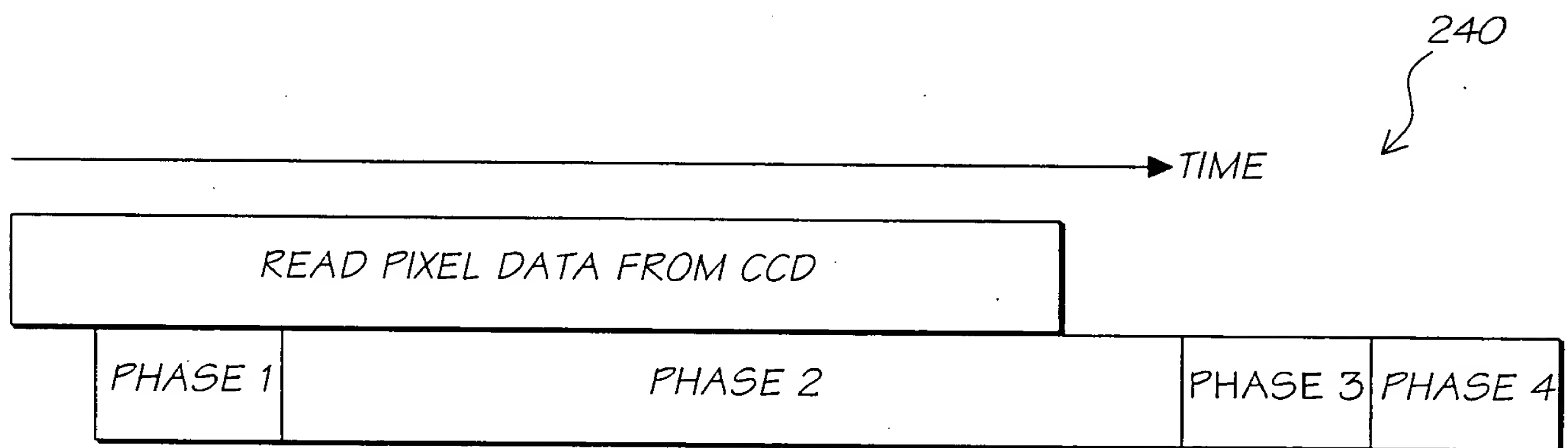


FIG. 34

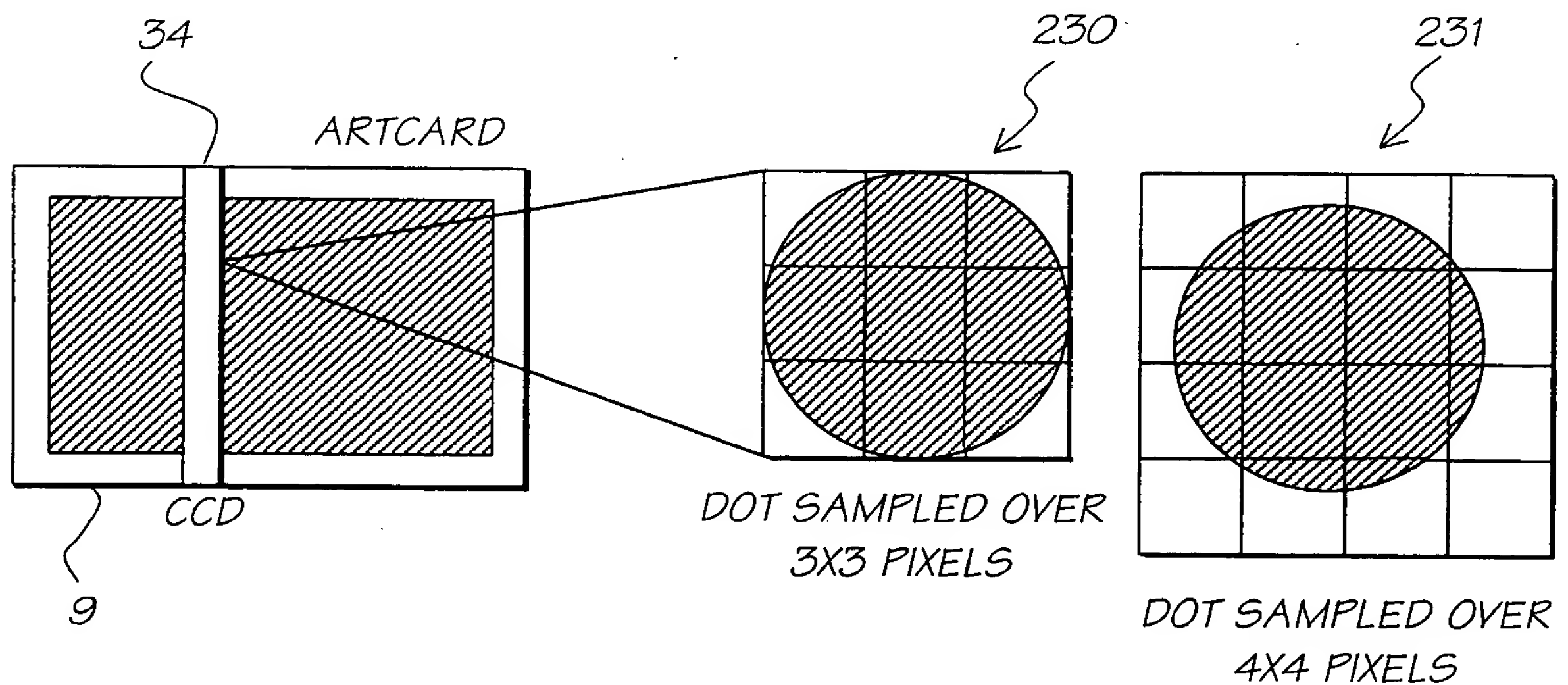


FIG. 35

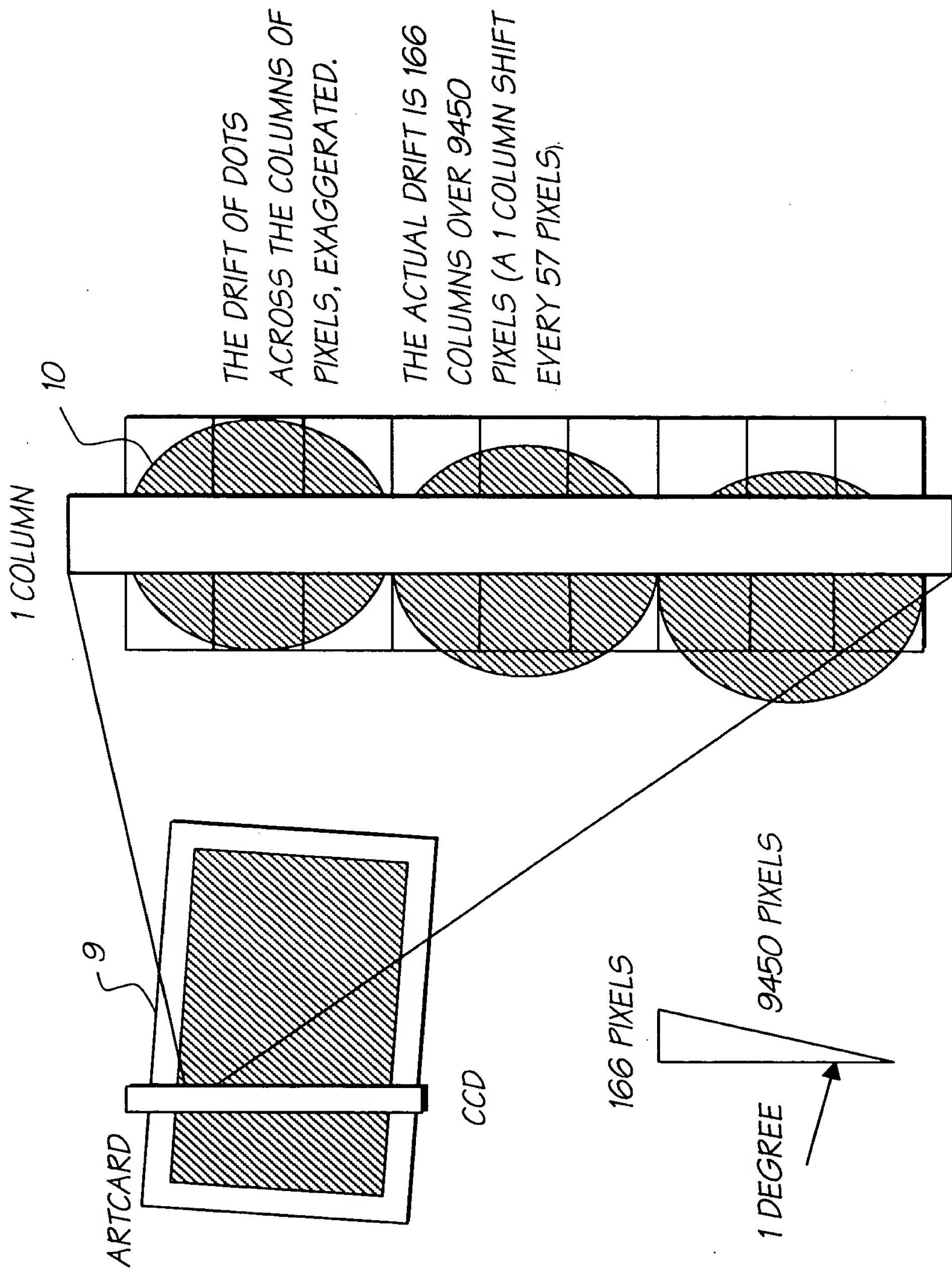


FIG. 36

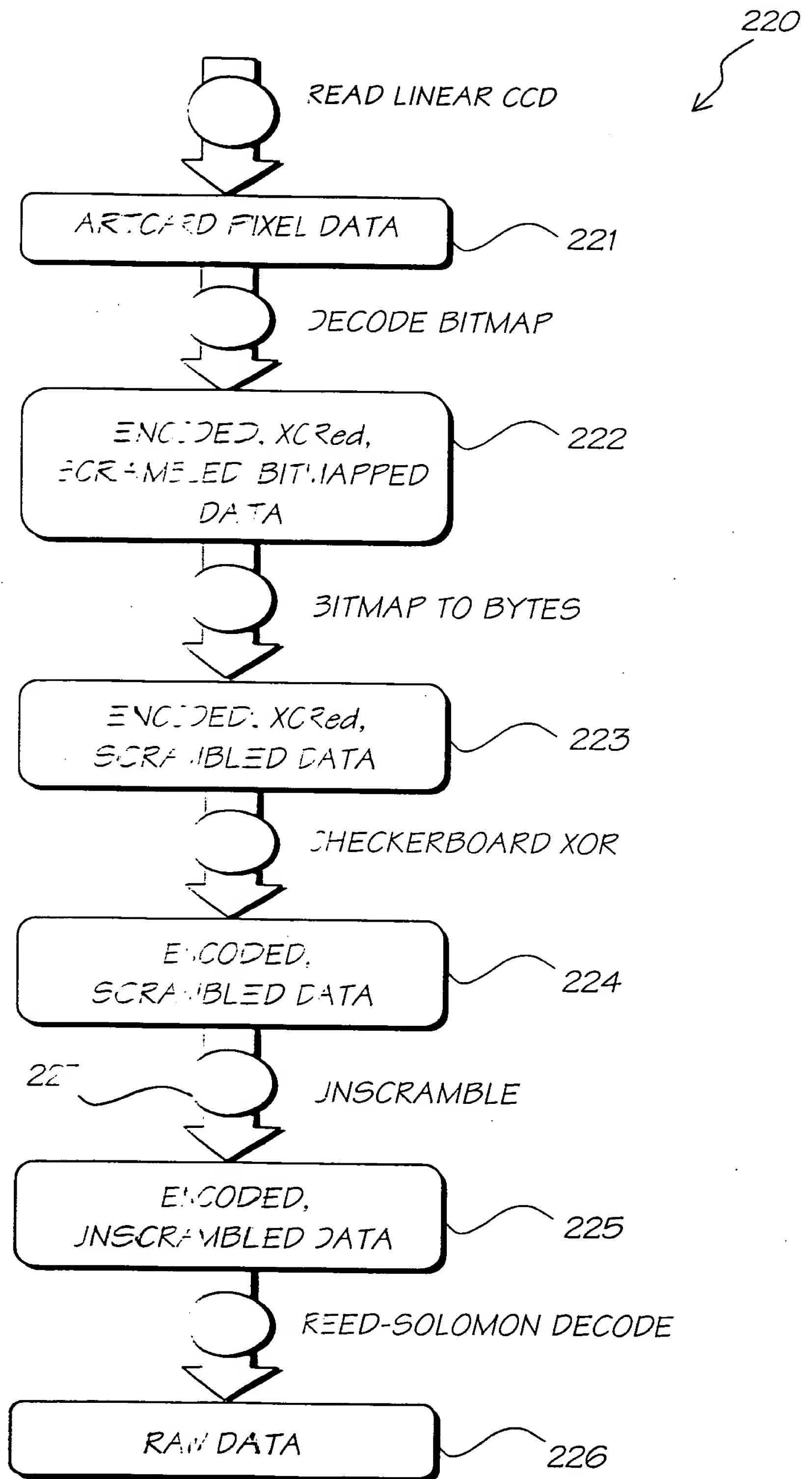


FIG. 37

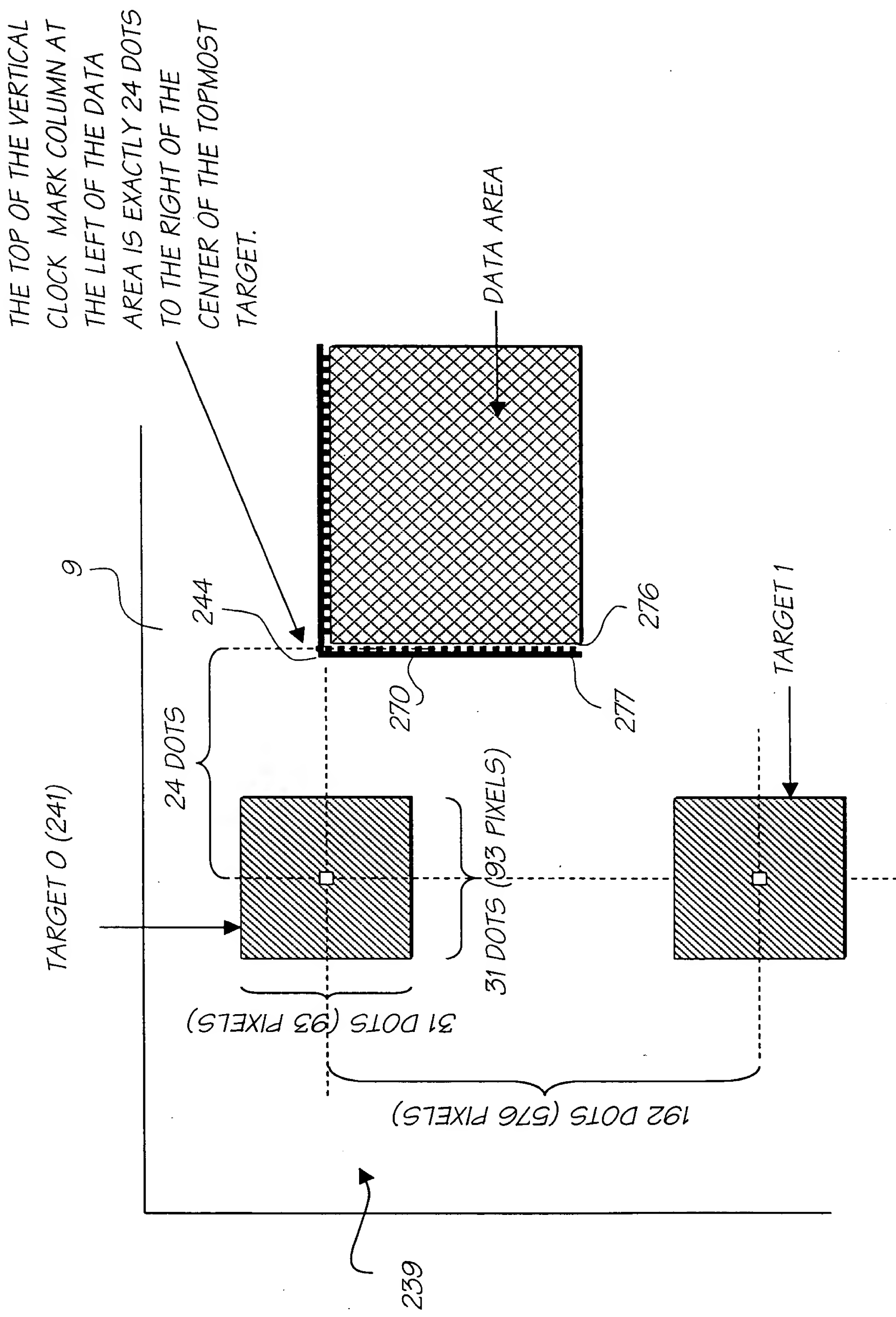


FIG. 38

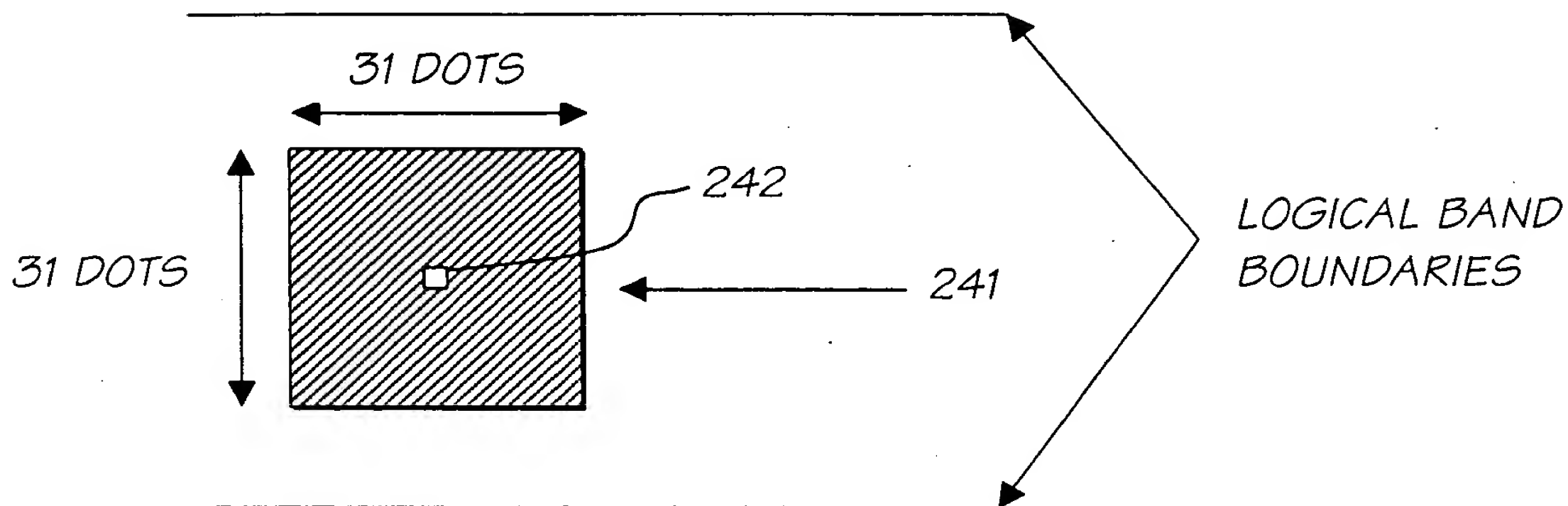


FIG. 39

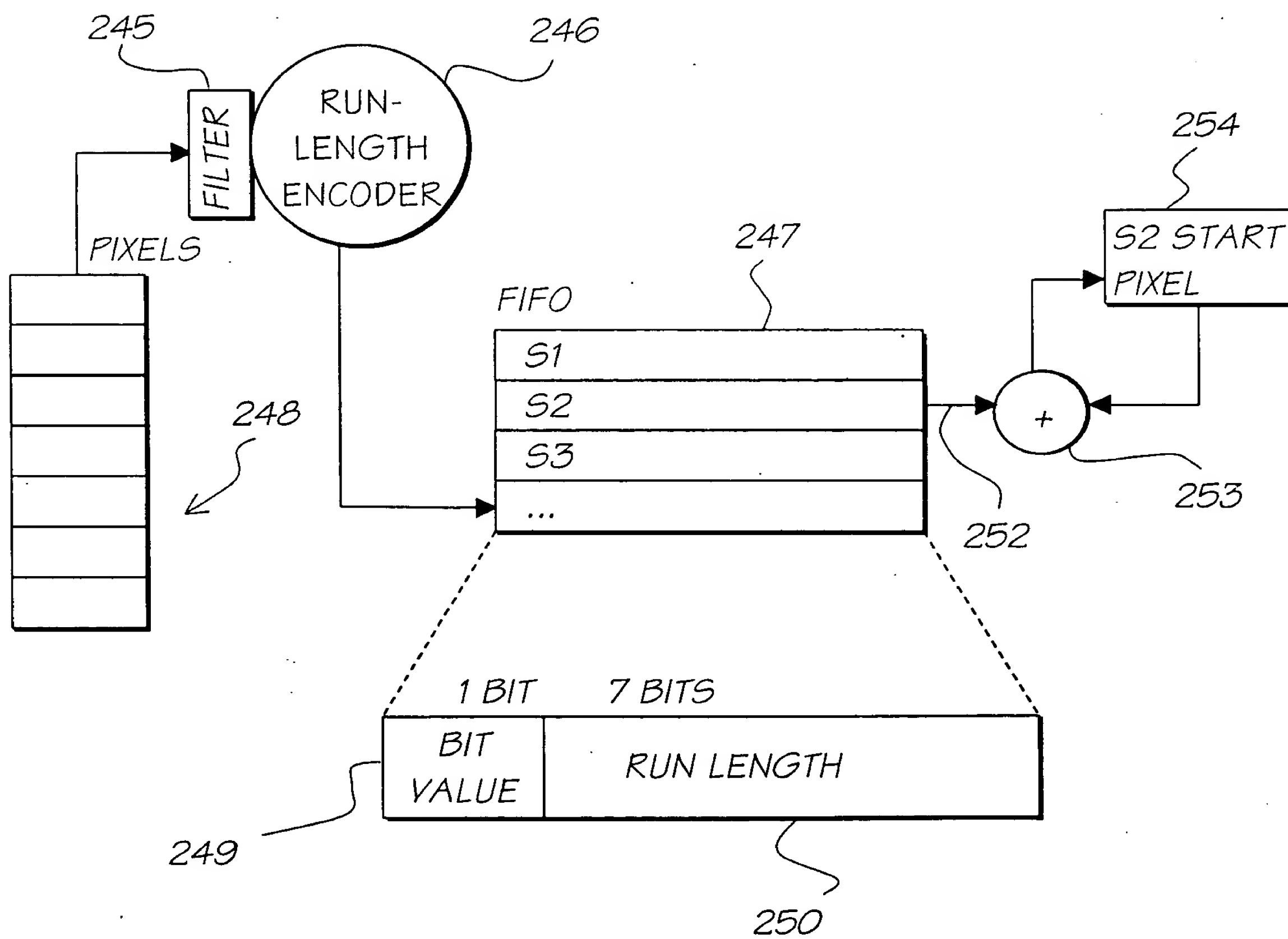


FIG. 40

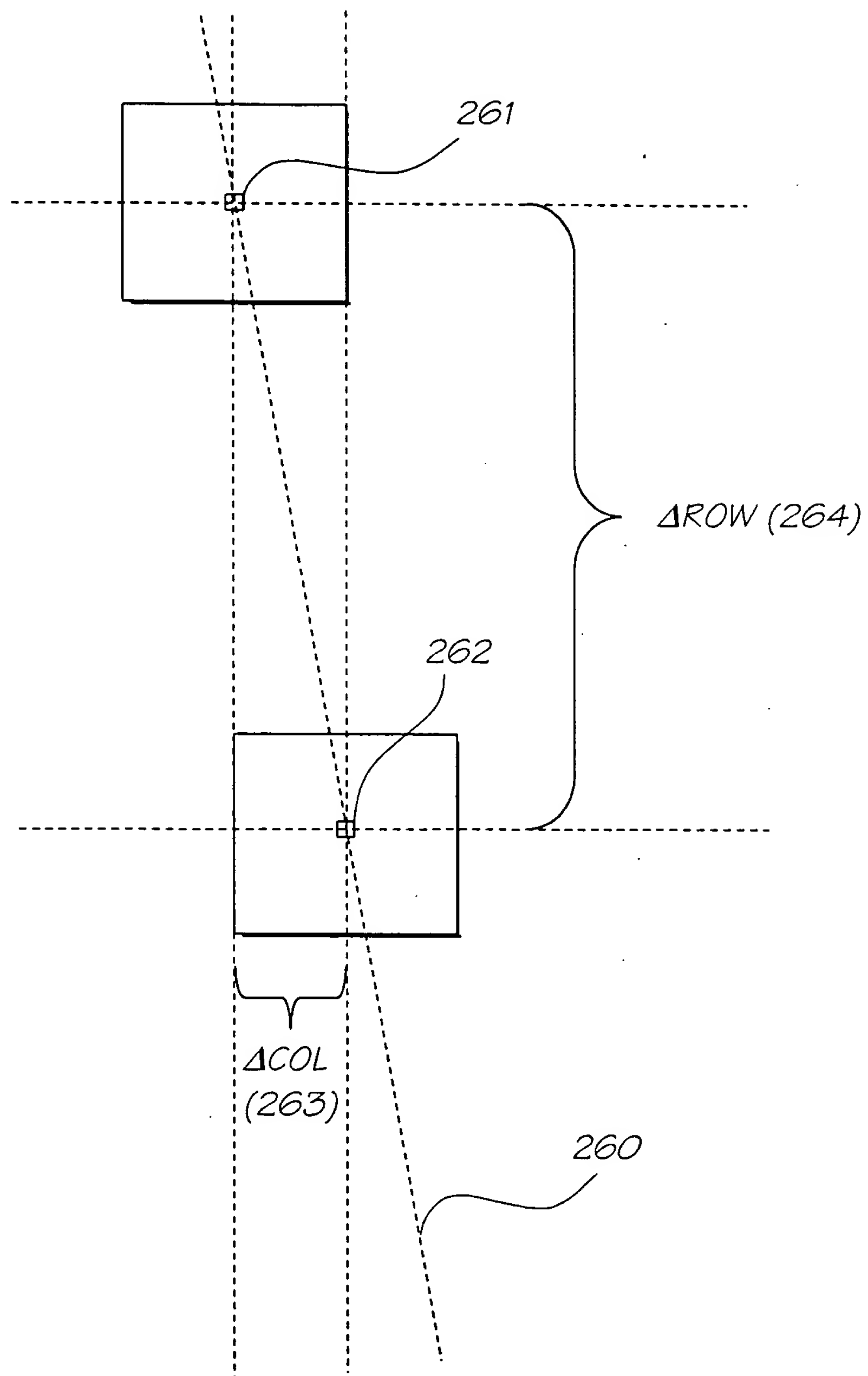


FIG. 41

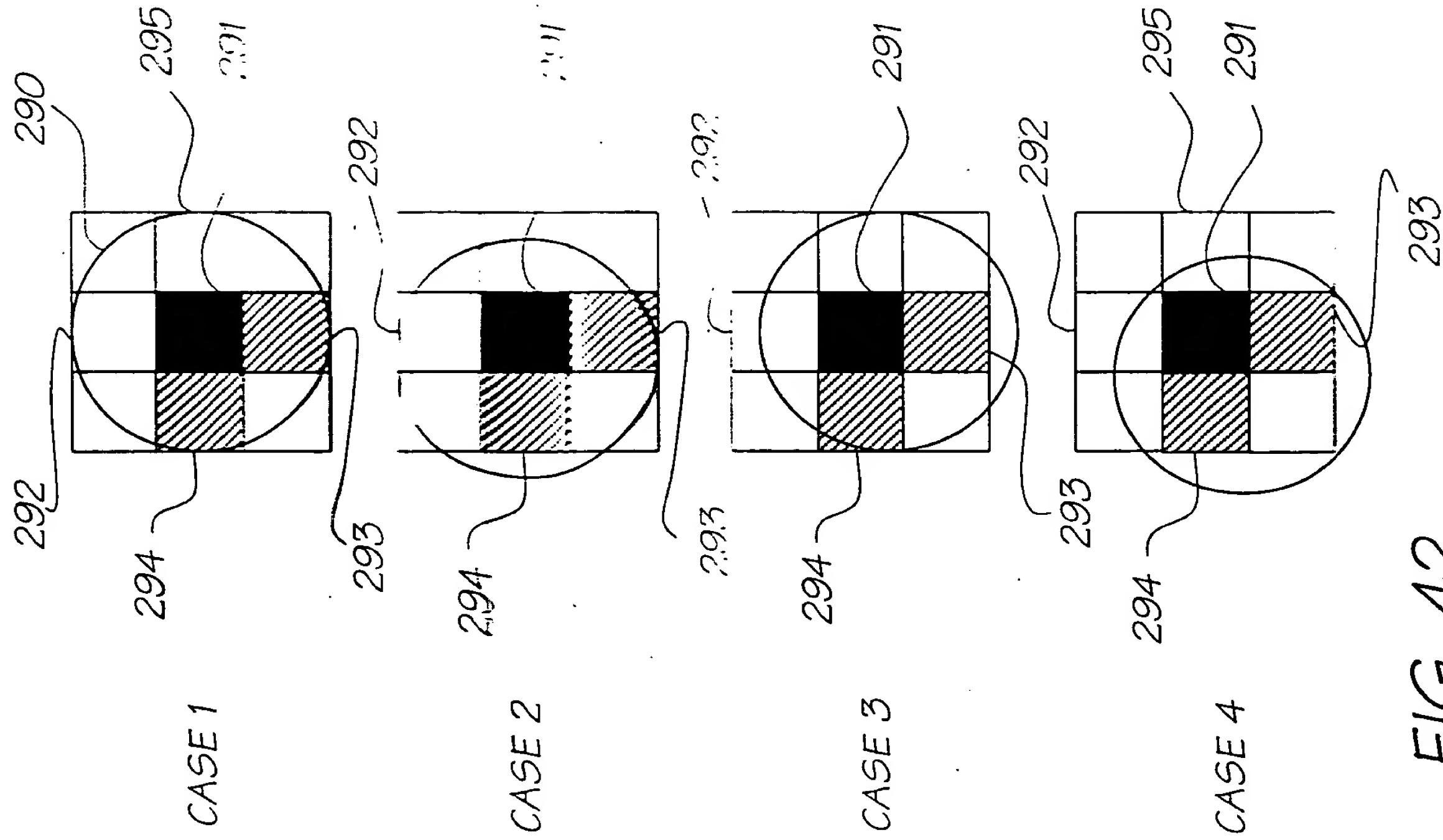


FIG. 42

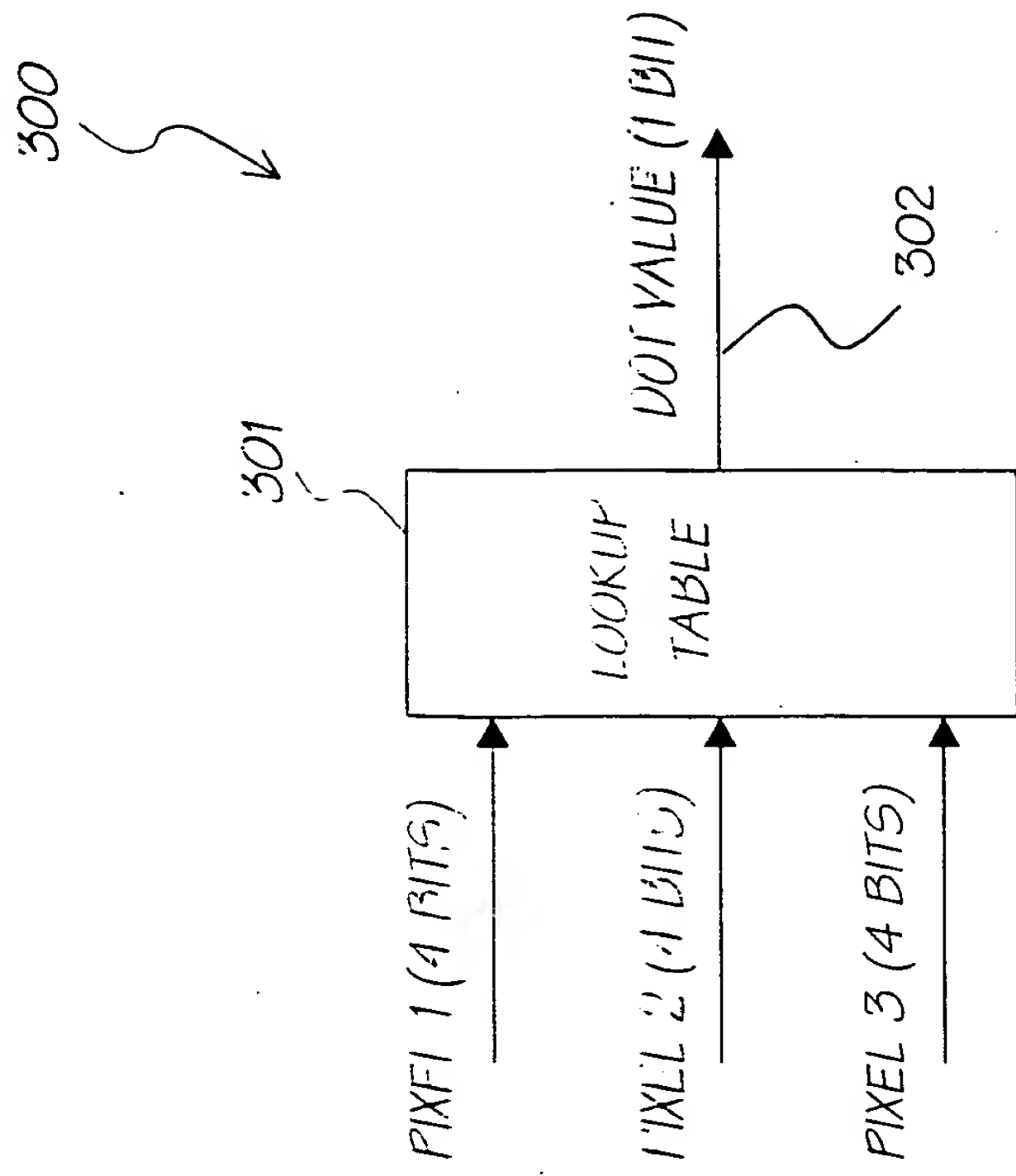


FIG. 43

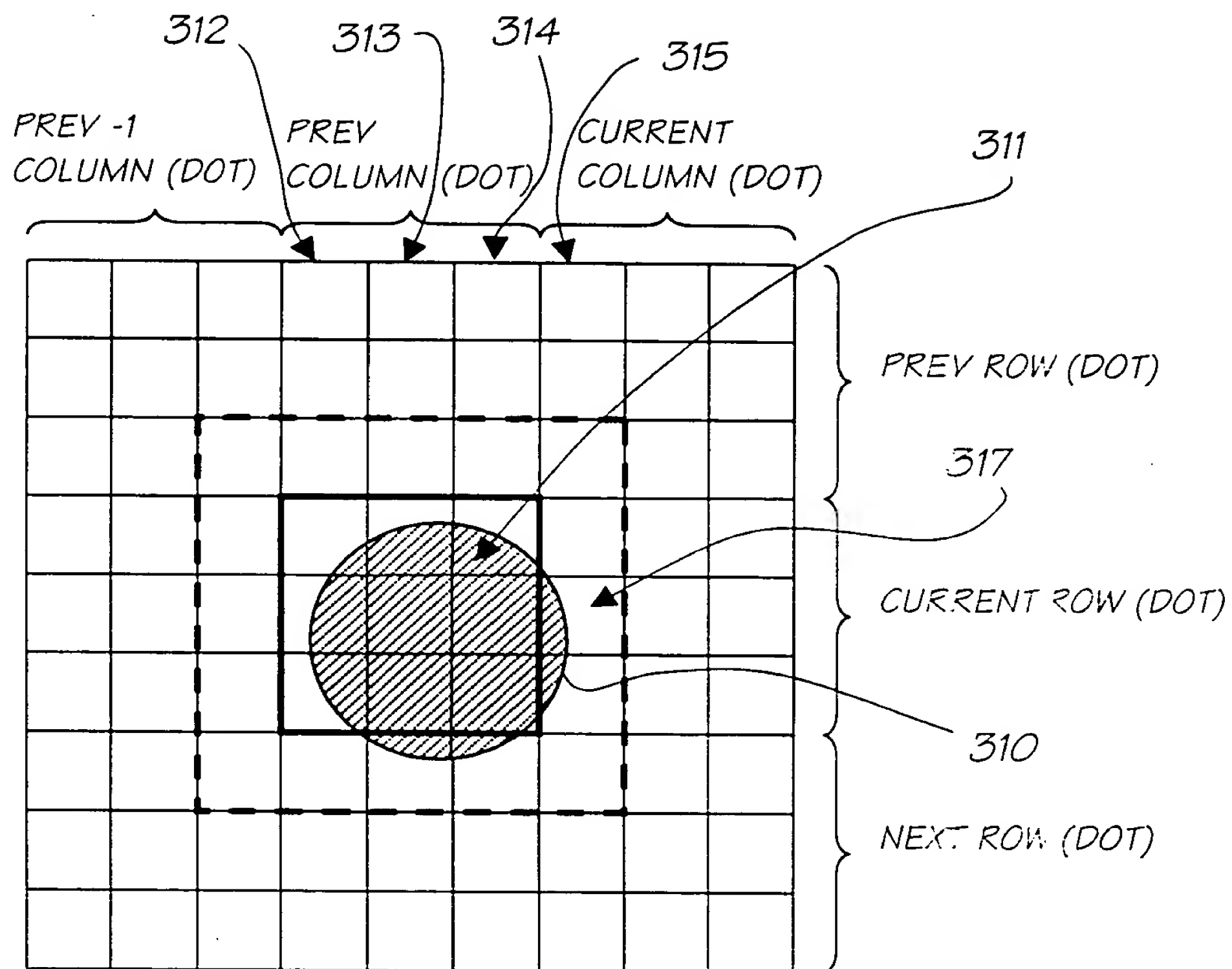


FIG. 44

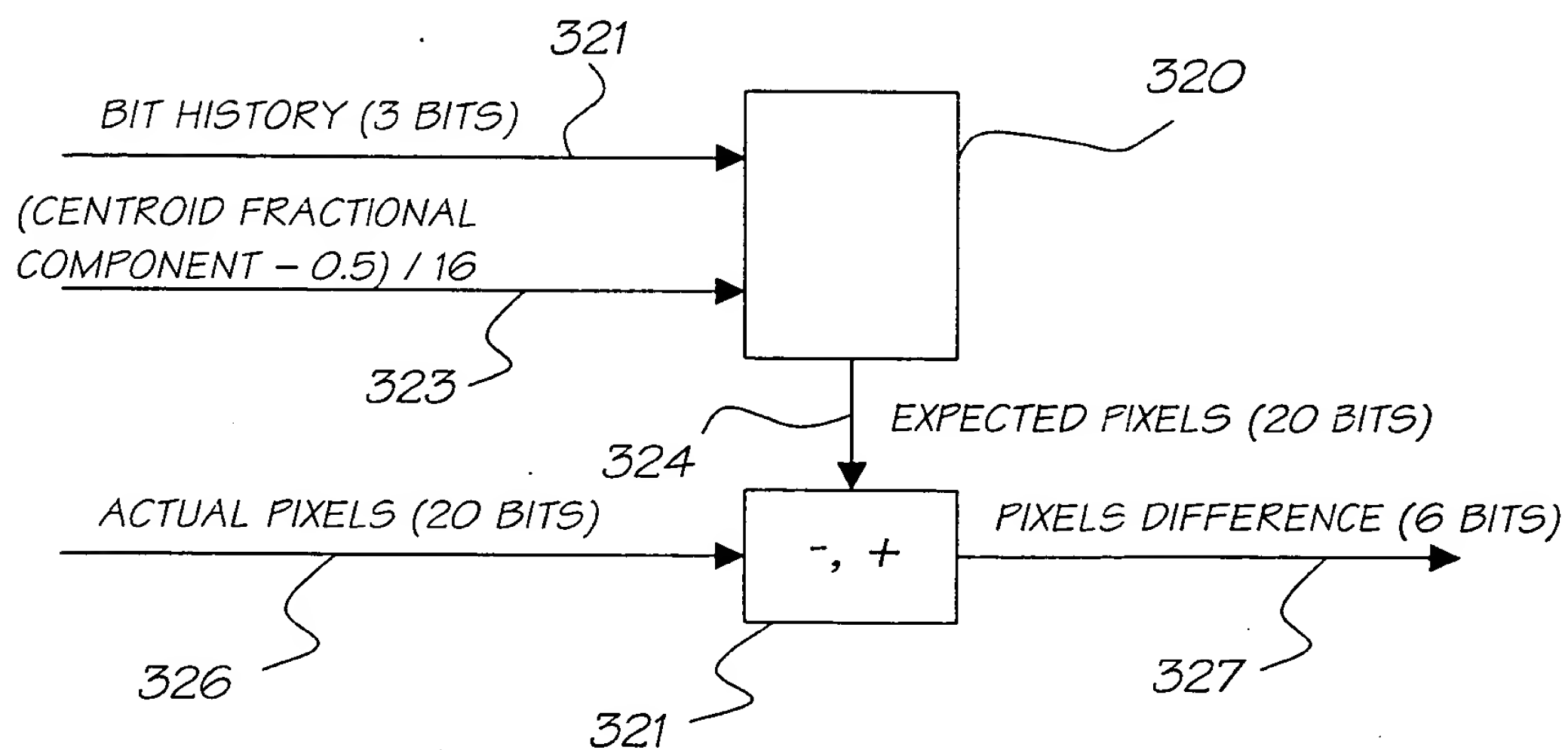


FIG. 45

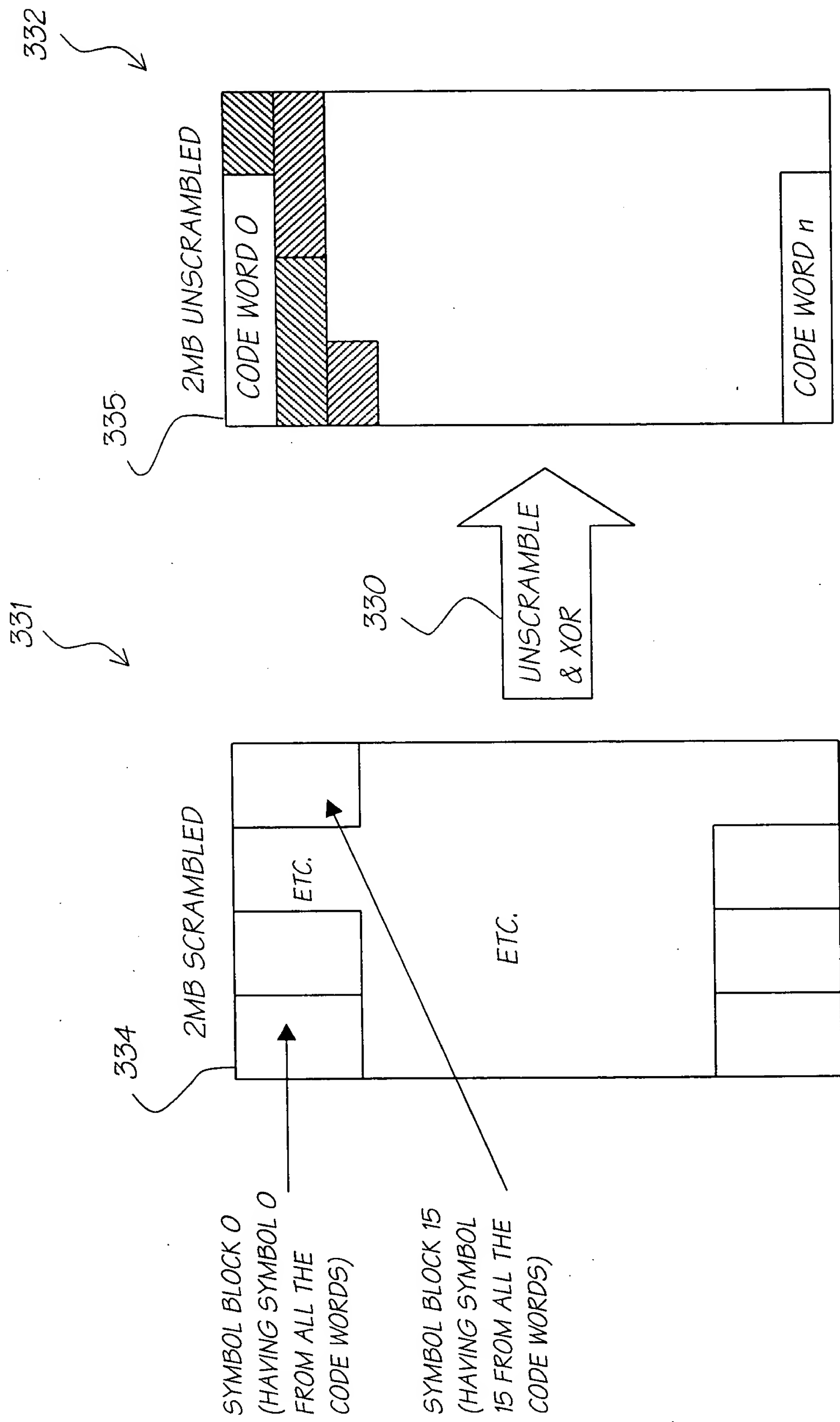
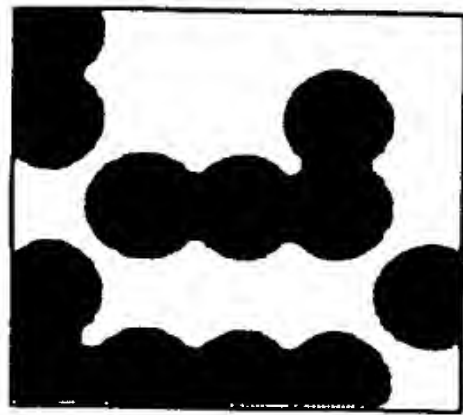
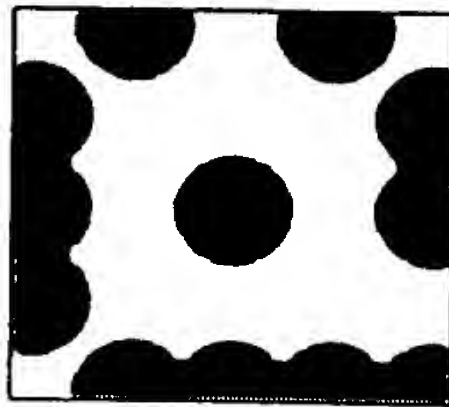


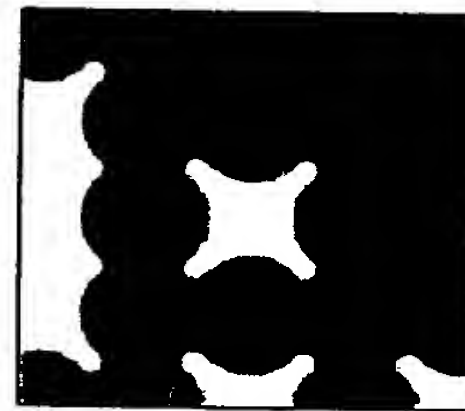
FIG. 46



BLACK AND WHITE
DOTS



BLACK DOT
SURROUNDED
BY WHITE



WHITE DOT
SURROUNDED
BY BLACK

FIG. 47

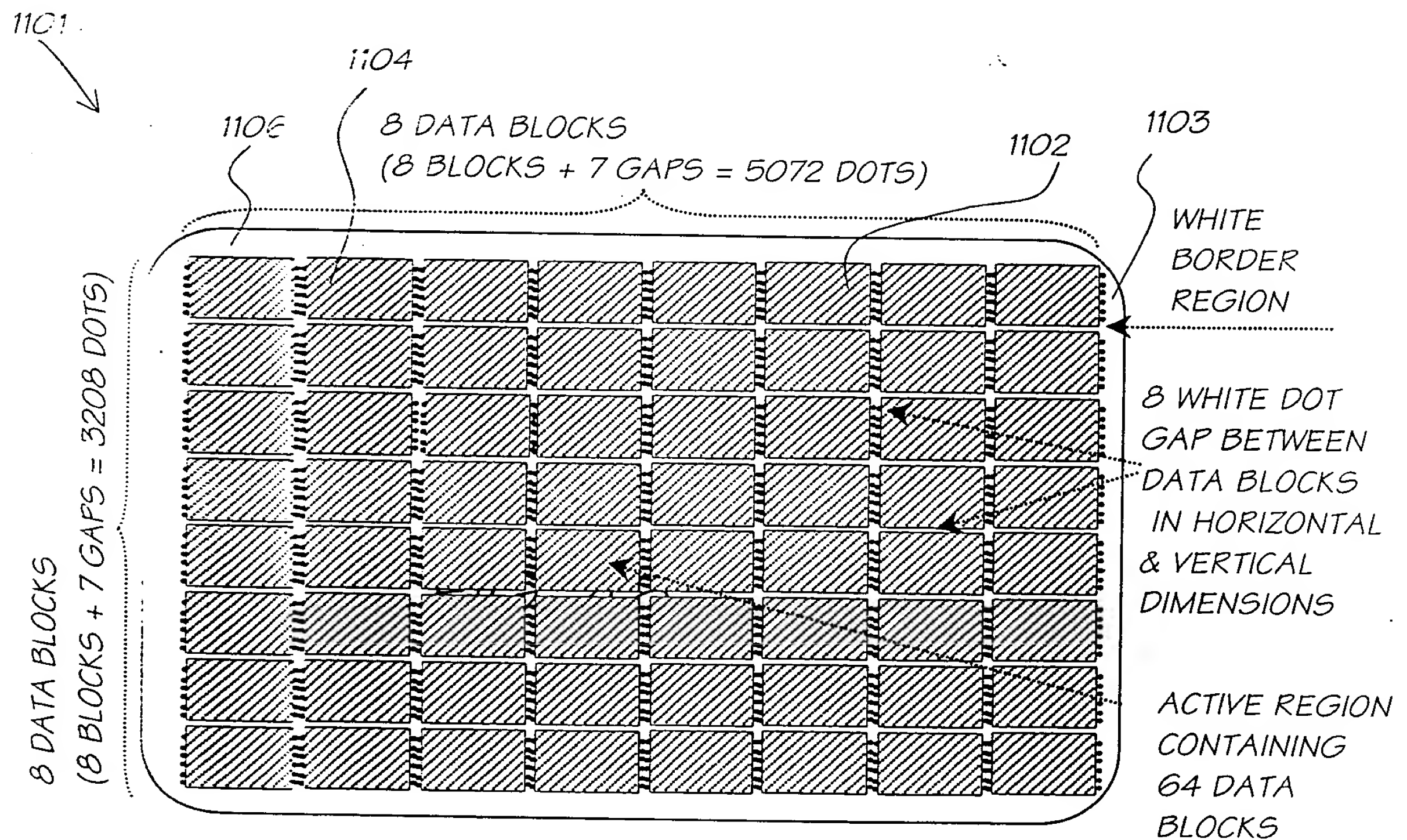


FIG. 48

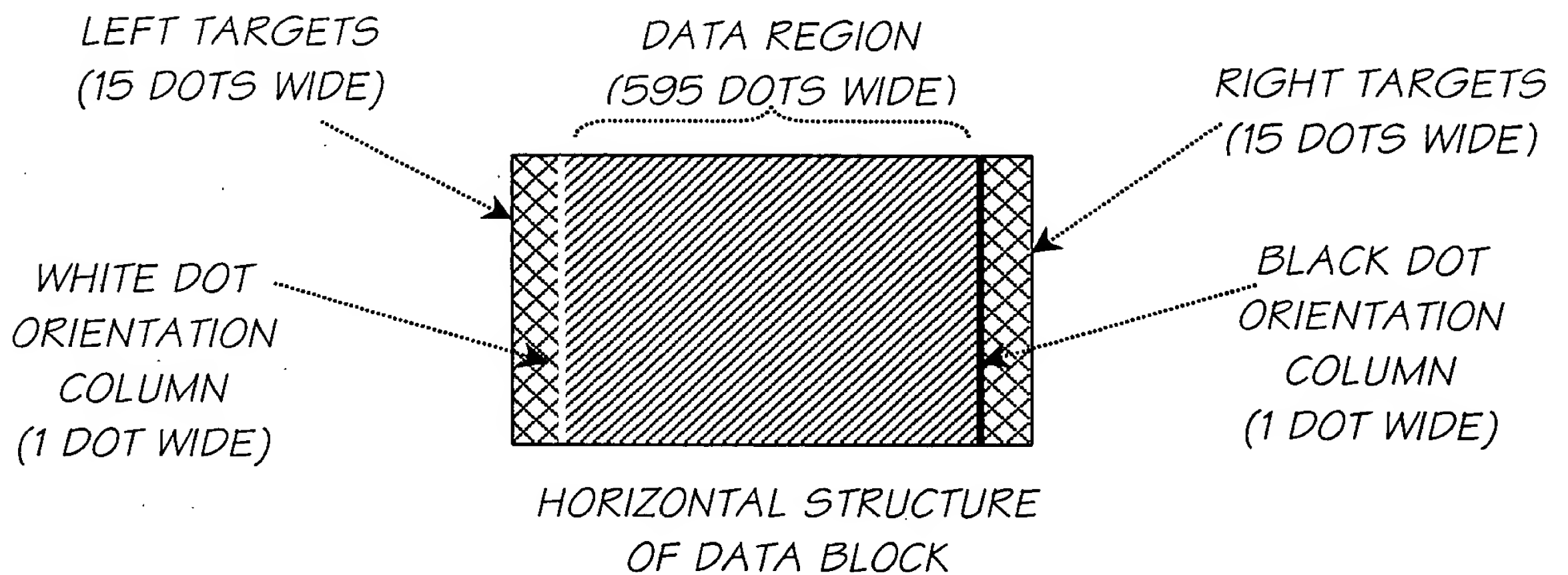
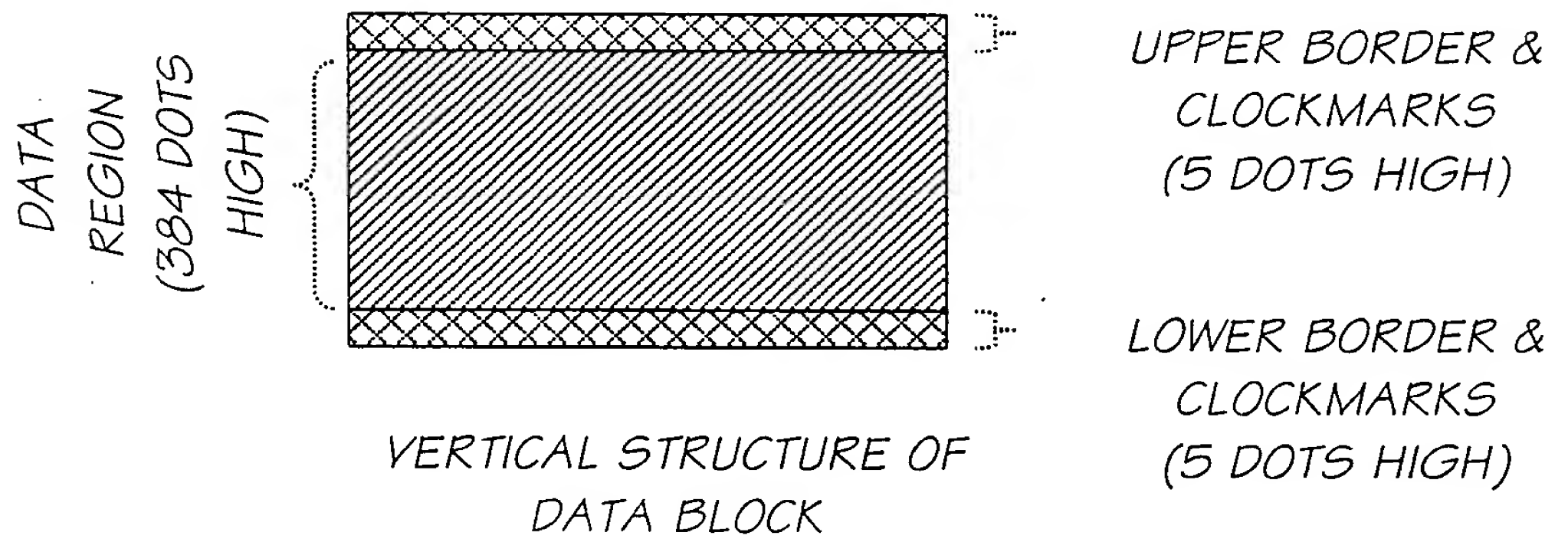
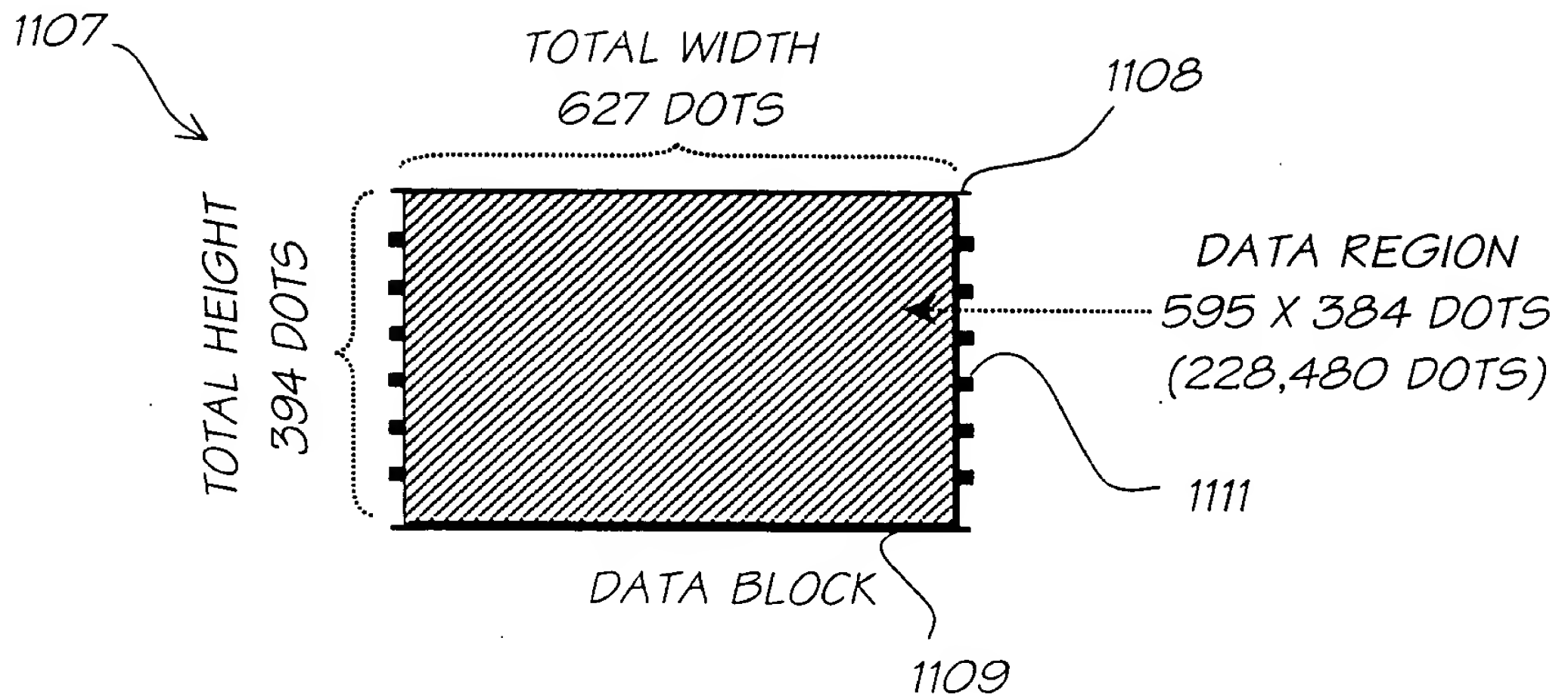


FIG. 49

FIG. 51

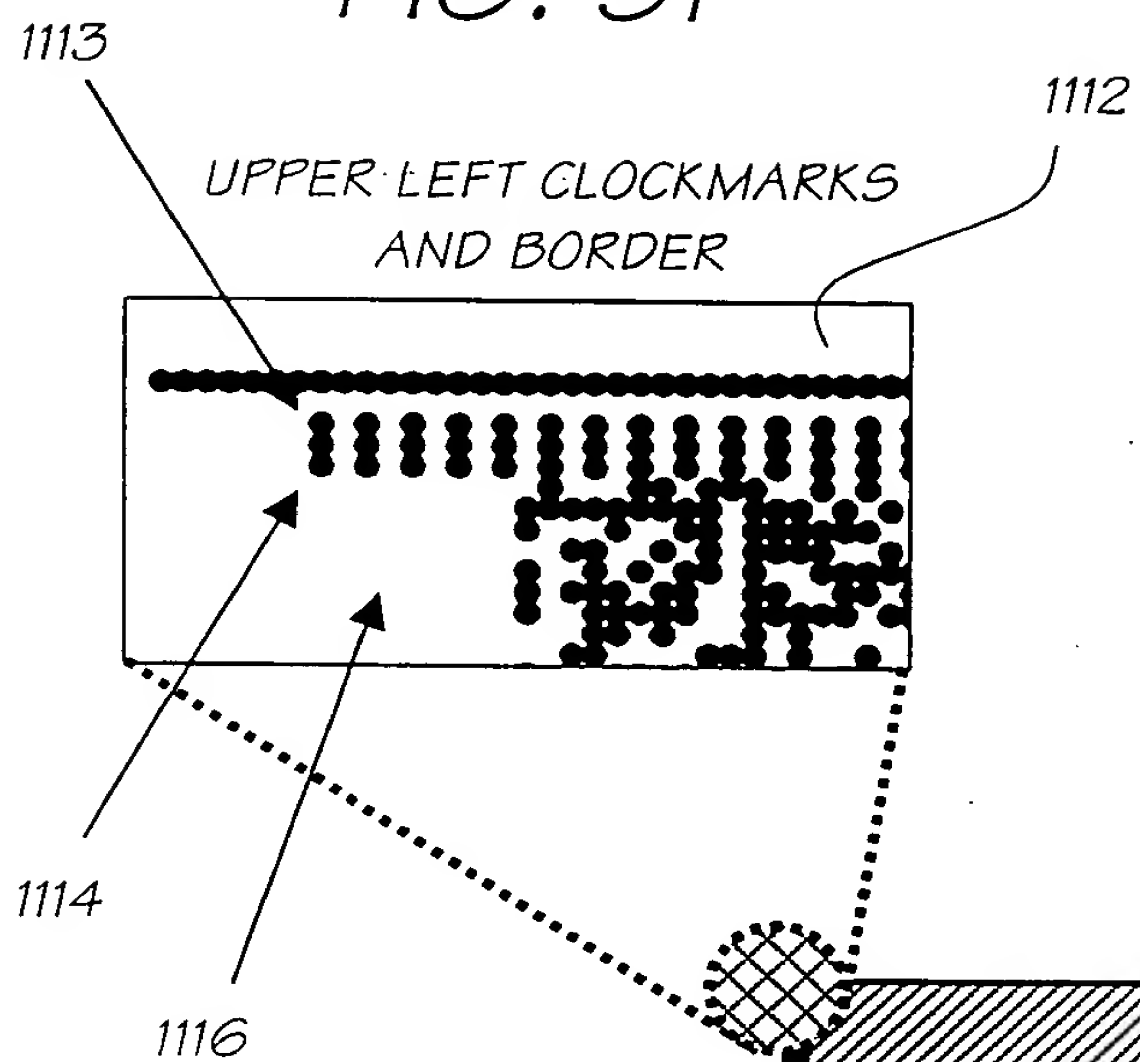


FIG. 52

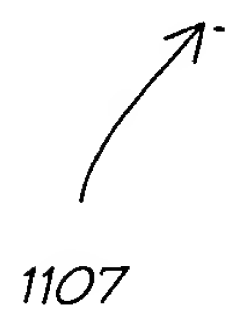
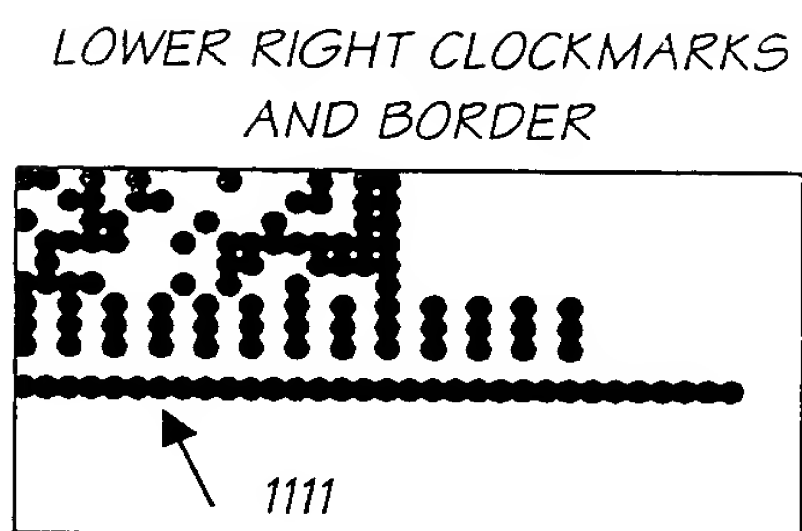


FIG. 50

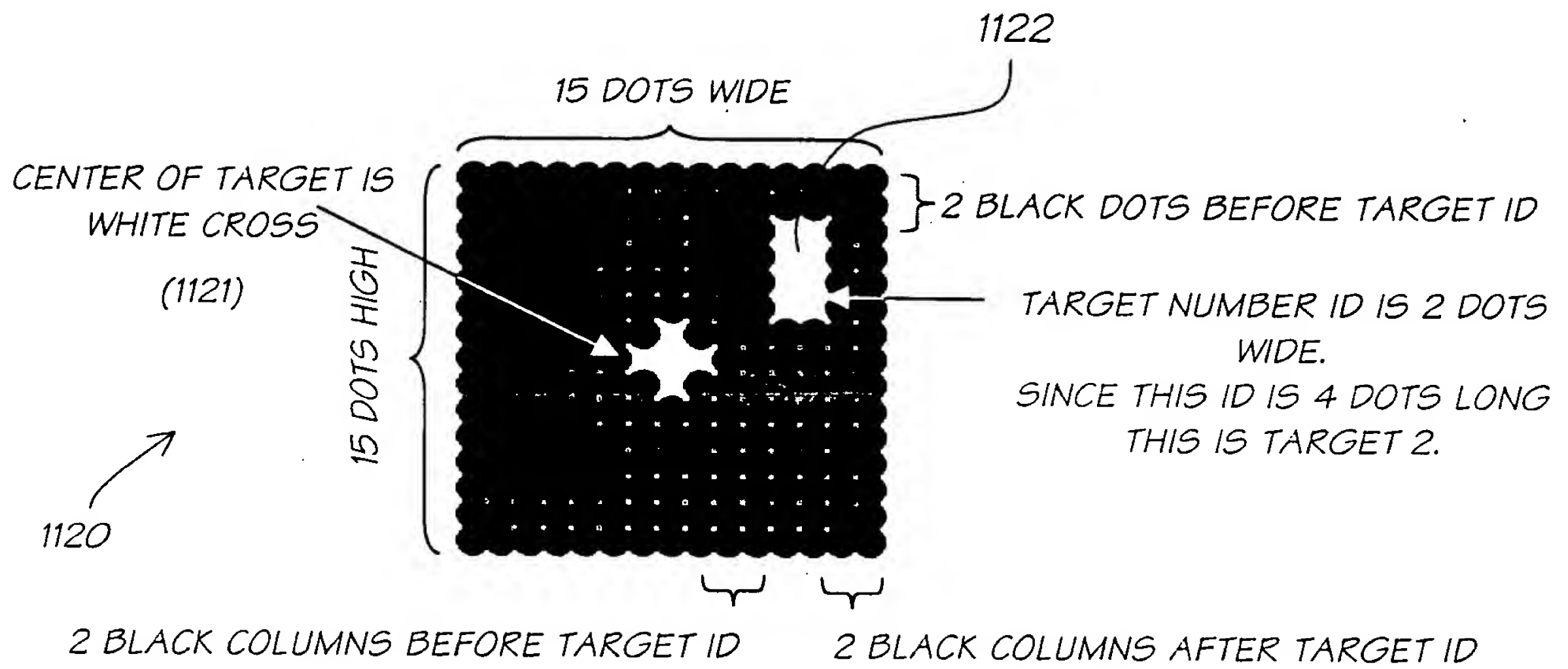


FIG. 53

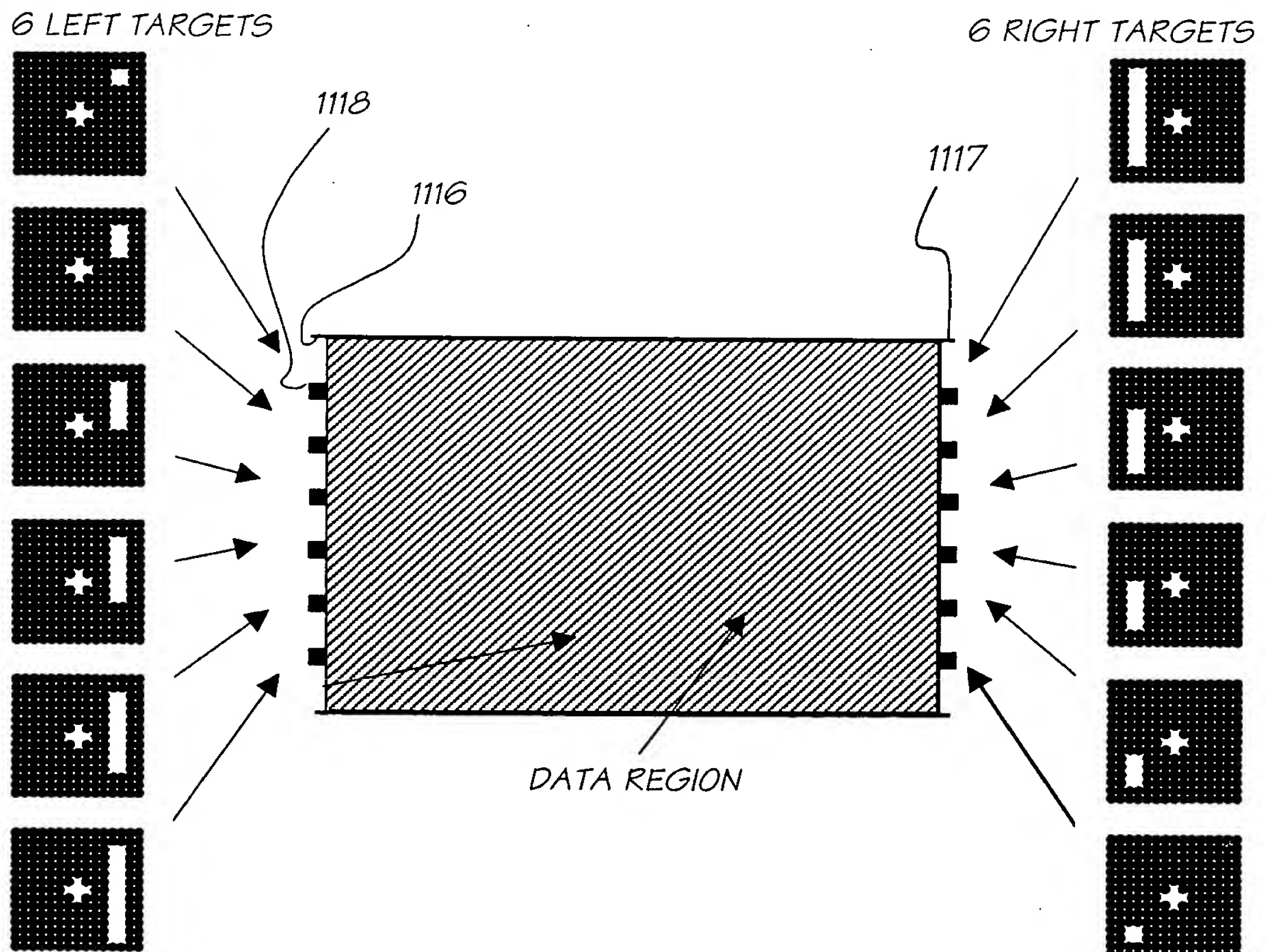


FIG. 54

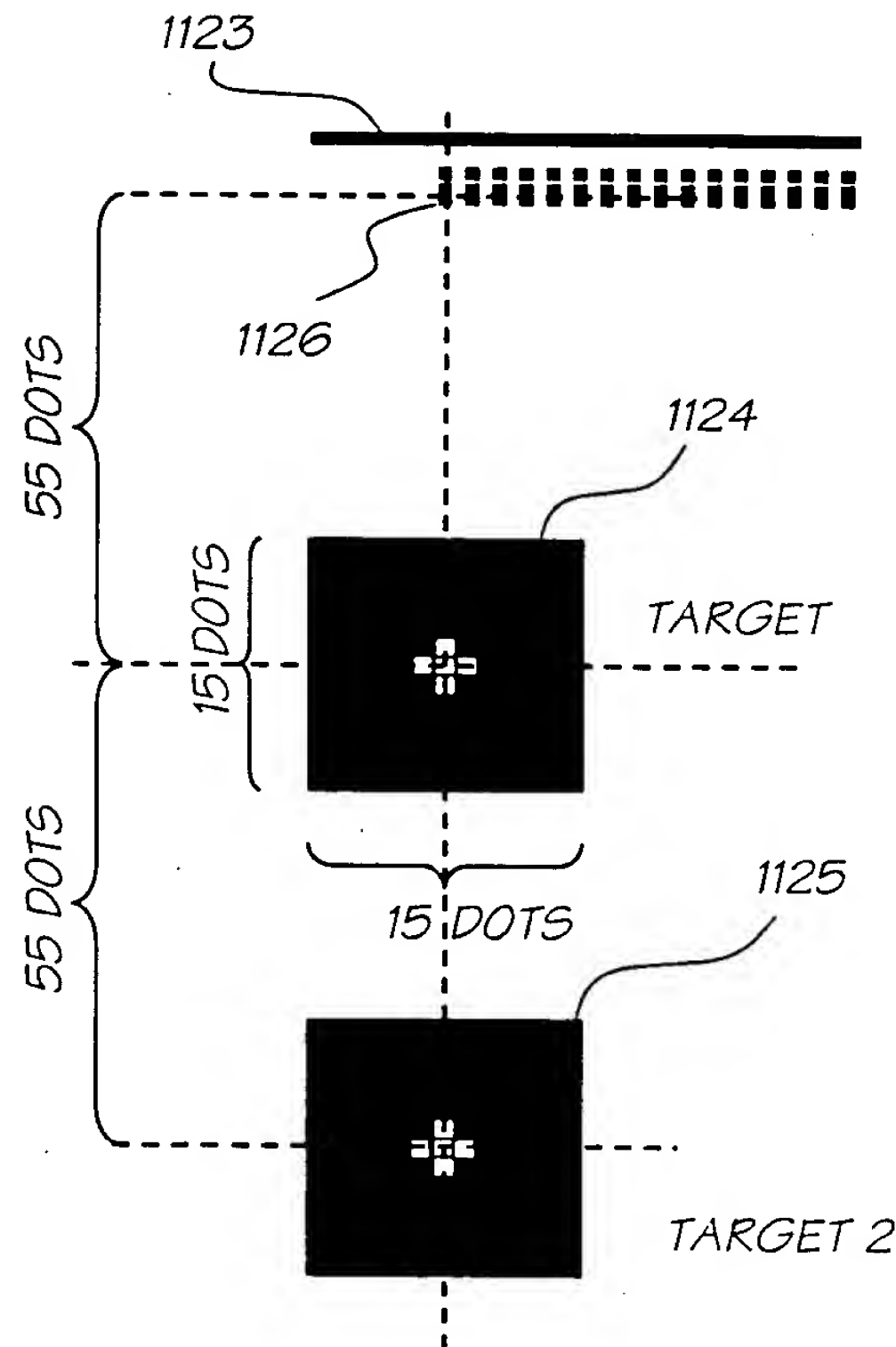


FIG. 55

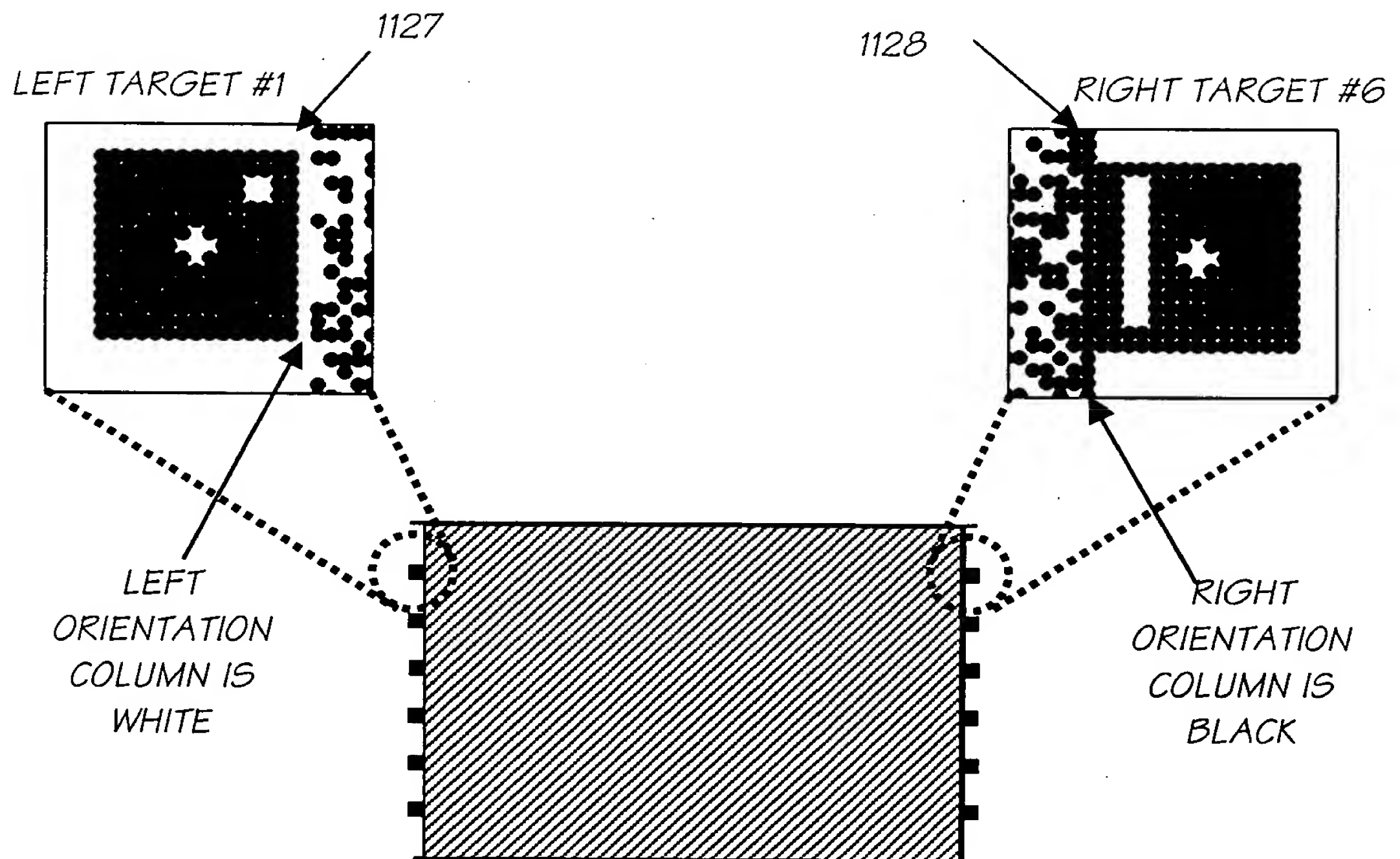


FIG. 56

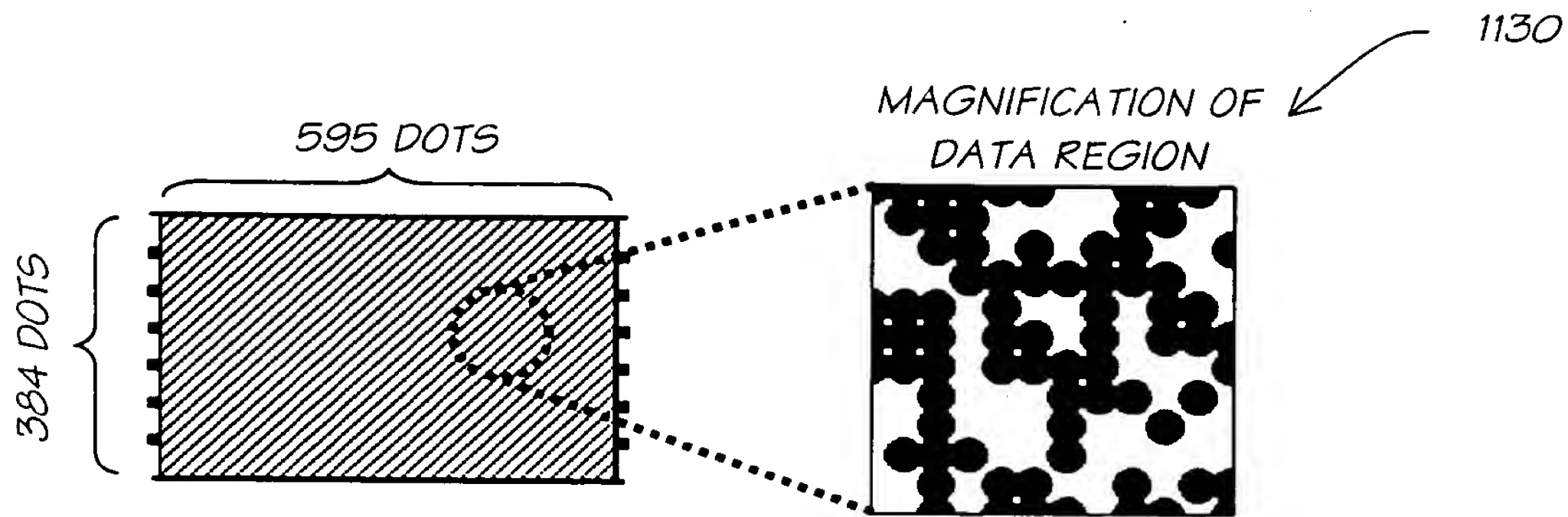


FIG. 57

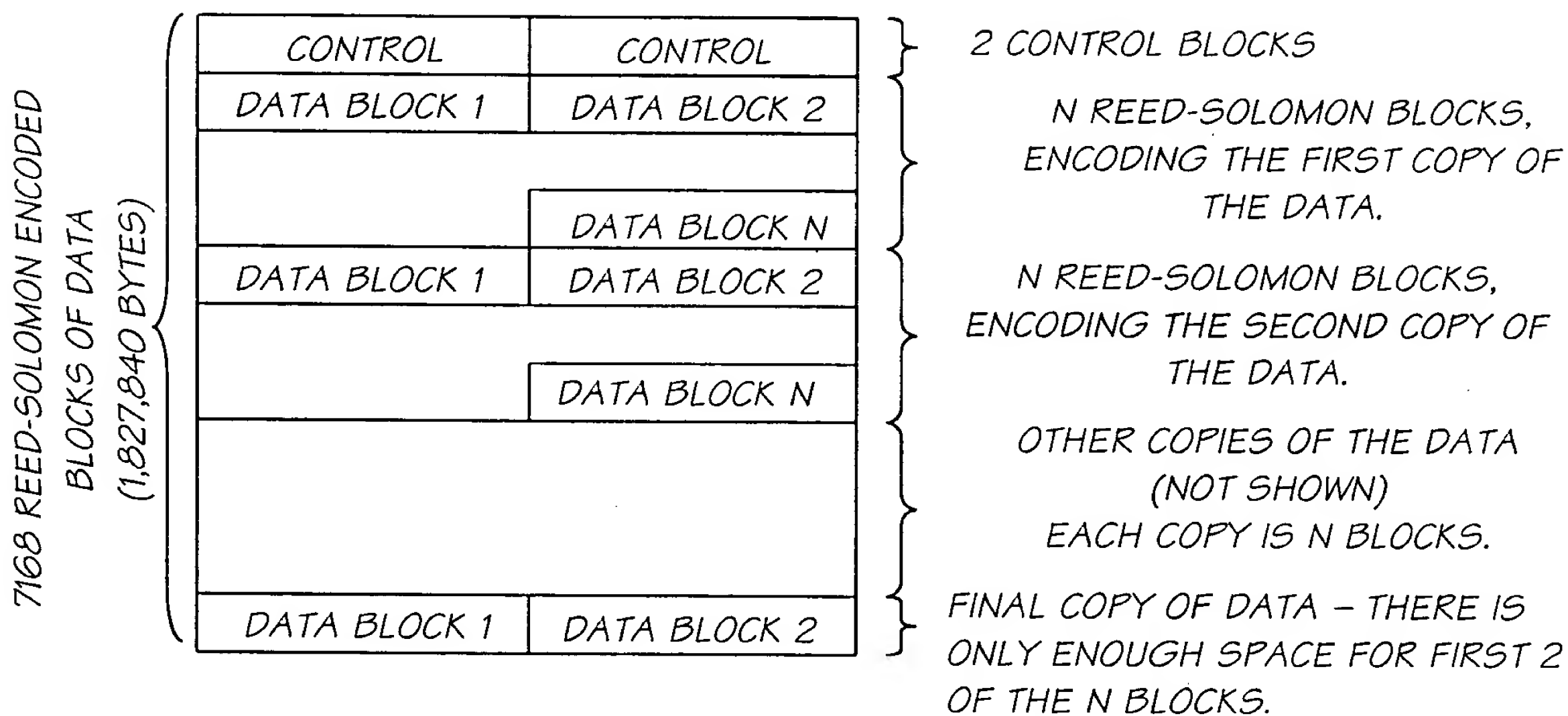


FIG. 58

00:	4F	00	3D	4F	00	3D	4F	00	3D	4F	00	3D
0C:	4F	00	3D	4F	00	3D	4F	00	3D	4F	00	3D
18:	4F	00	3D	4F	00	3D	4F	00	3D	4F	00	3D
24:	4F	00	3D	4F	00	3D	4F	00	3D	4F	00	3D
30:	4F	00	3D	4F	00	3D	4F	00	3D	4F	00	3D
3C:	4F	00	3D	4F	00	3D	4F	00	3D	4F	00	3D
48:	4F	00	3D	4F	00	3D	4F	00	3D	4F	00	3D
54:	4F	00	3D	4F	00	3D	4F	00	3D	4F	00	3D
60:	00	00	00	00	00	00	00	00	00	00	00	00
6C:	00	00	00	00	00	00	00	00	00	00	00	00
78:	00	00	00	00	00	00	00	00	00	00	00	00

32 COPIES OF THE 3 BYTE CONTROL INFORMATION

RESERVED BYTES ARE 0

FIG. 59

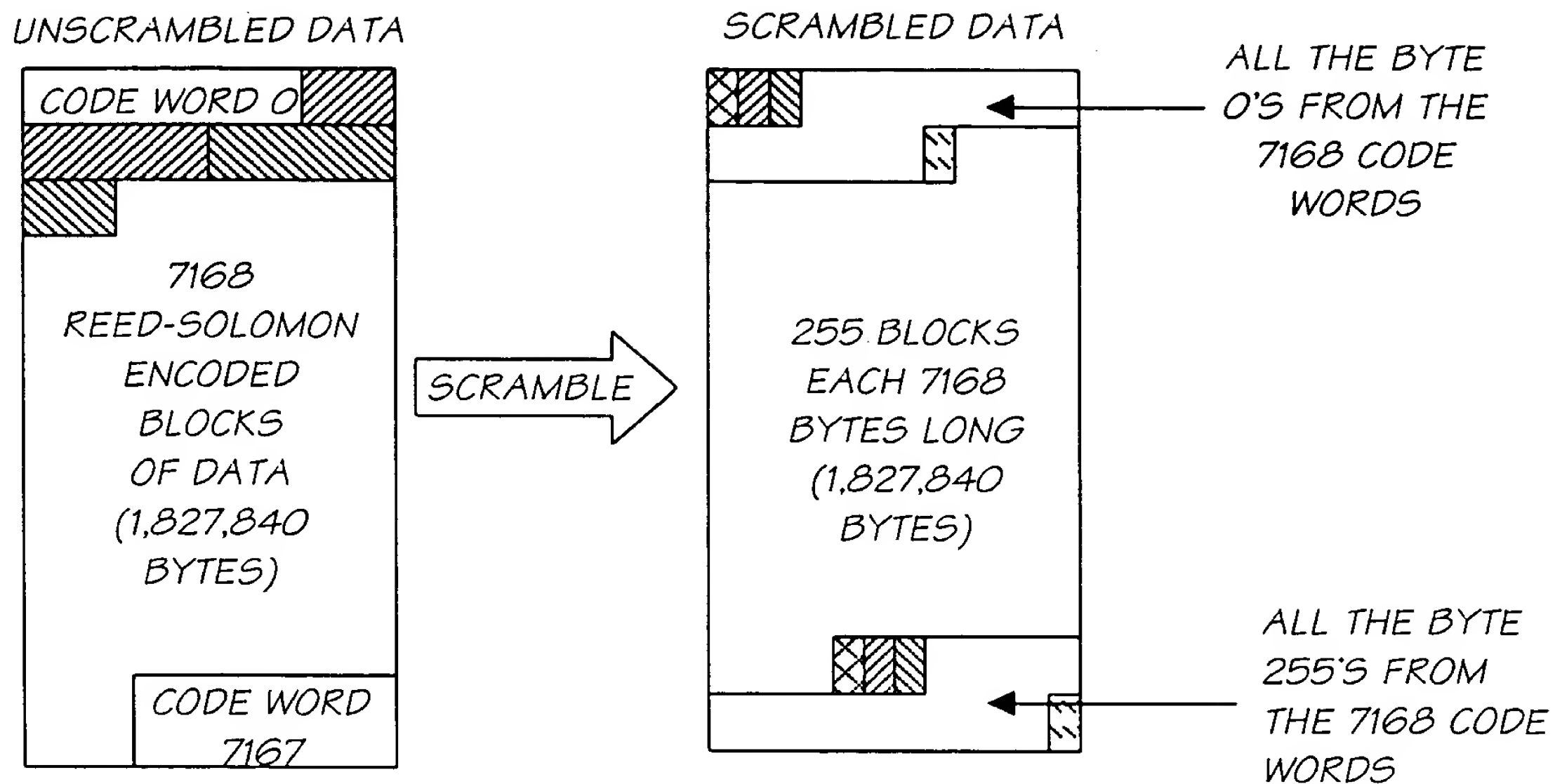


FIG. 60

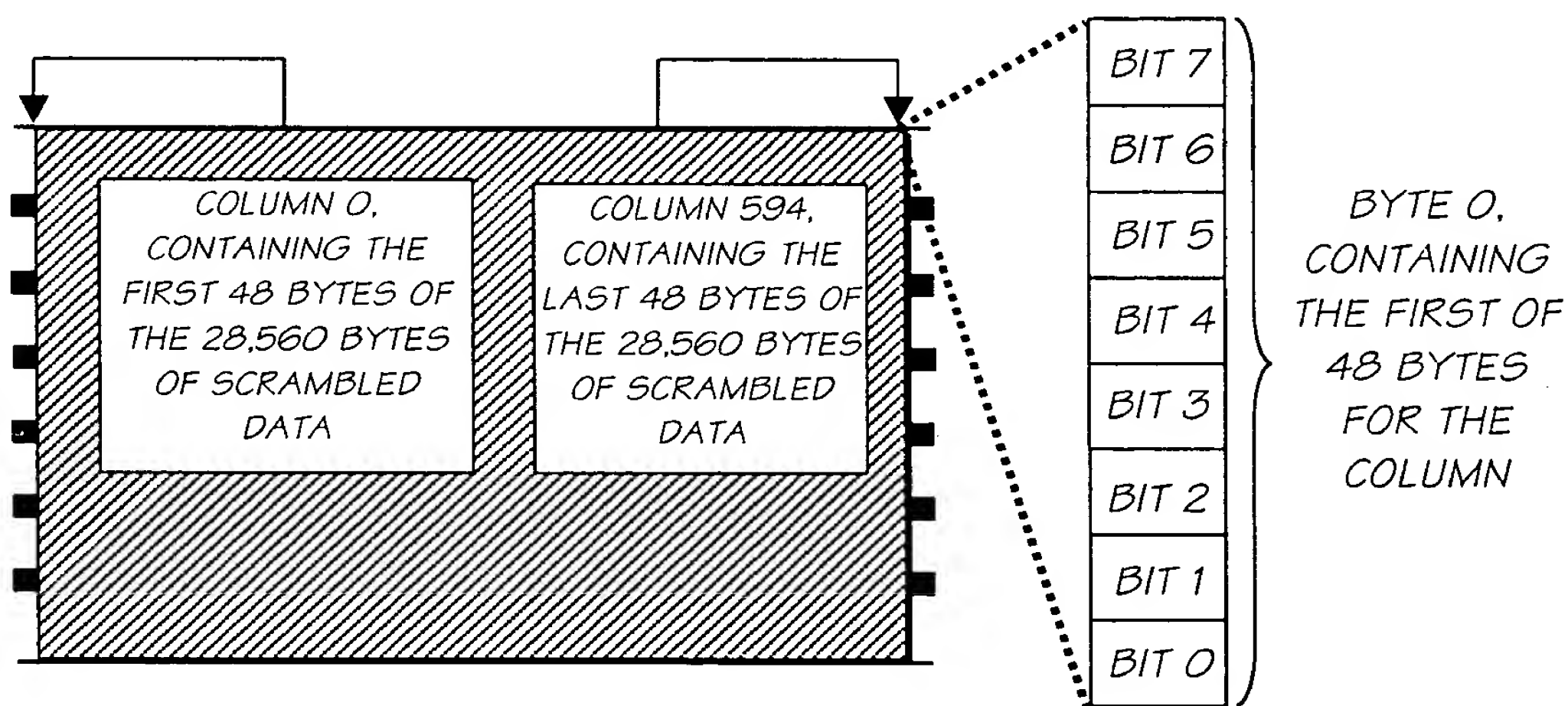


FIG. 61

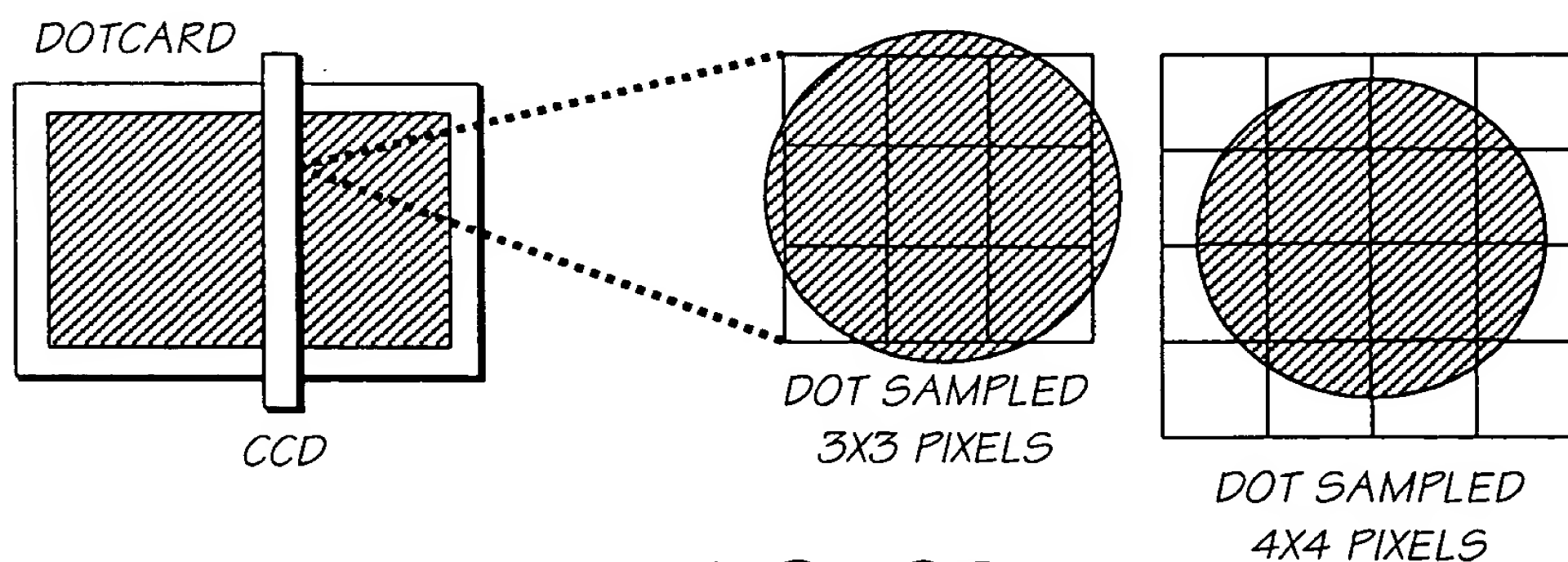


FIG. 62

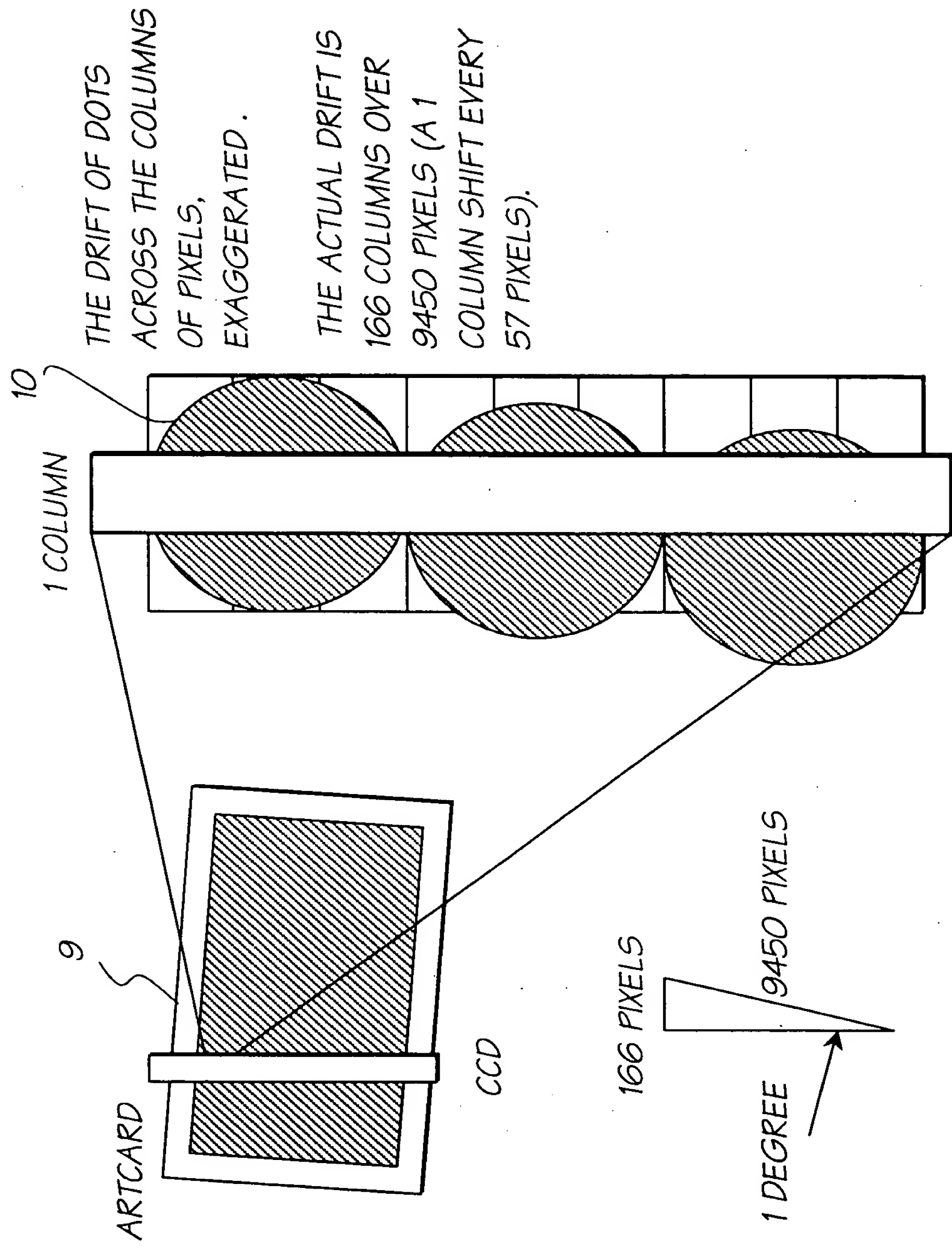


FIG. 63

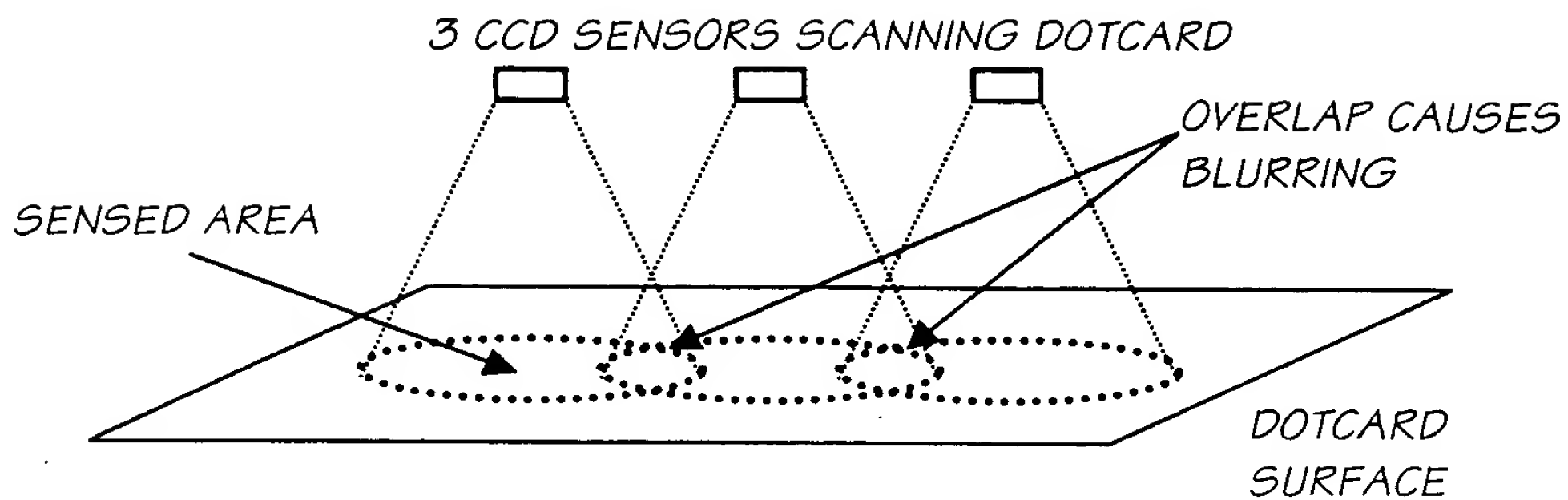


FIG. 64

RANGE OF BLACK DOTS
(FREQUENCY DISTRIBUTION)

RANGE OF WHITE DOTS
(FREQUENCY DISTRIBUTION)

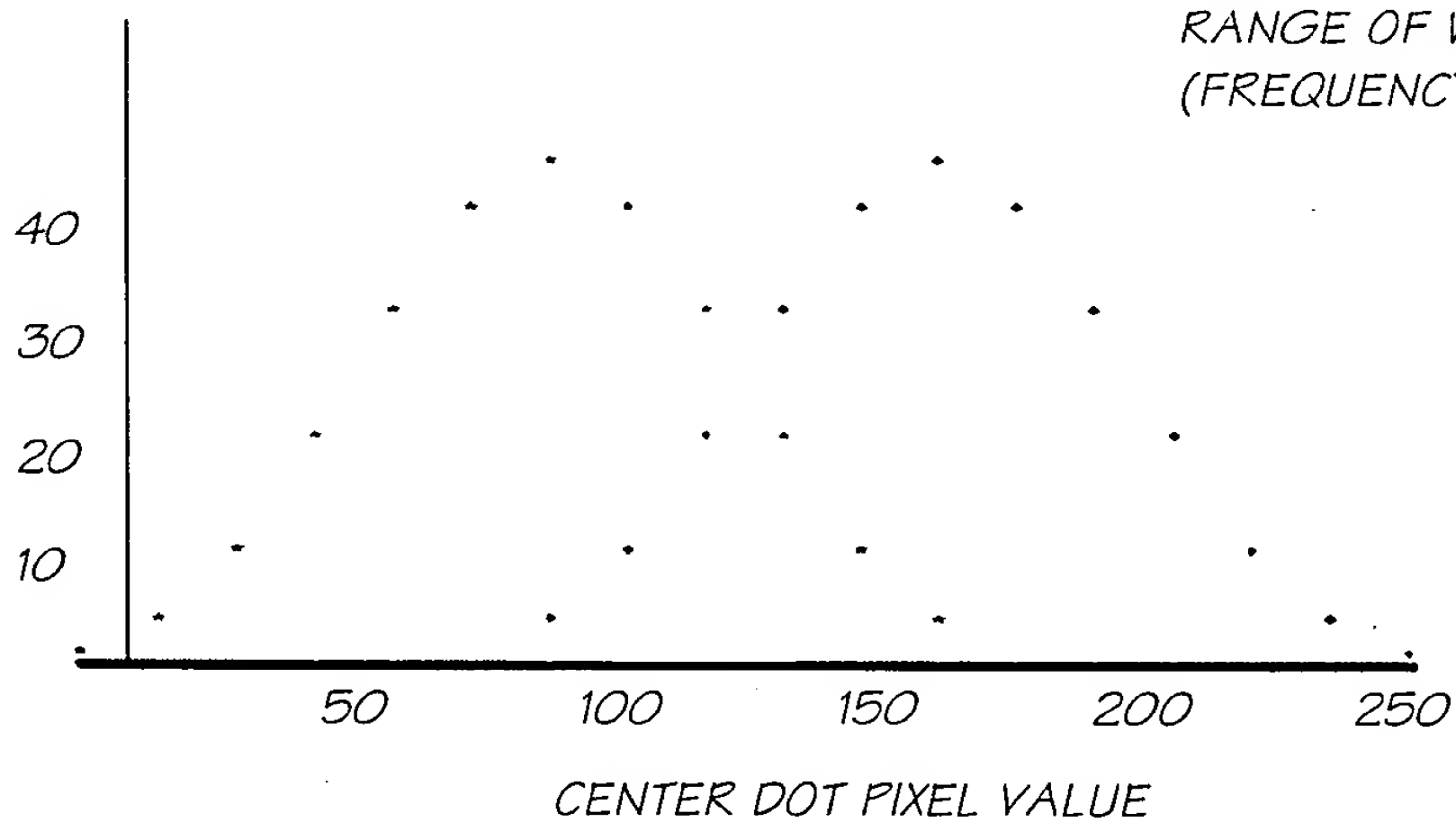


FIG. 65

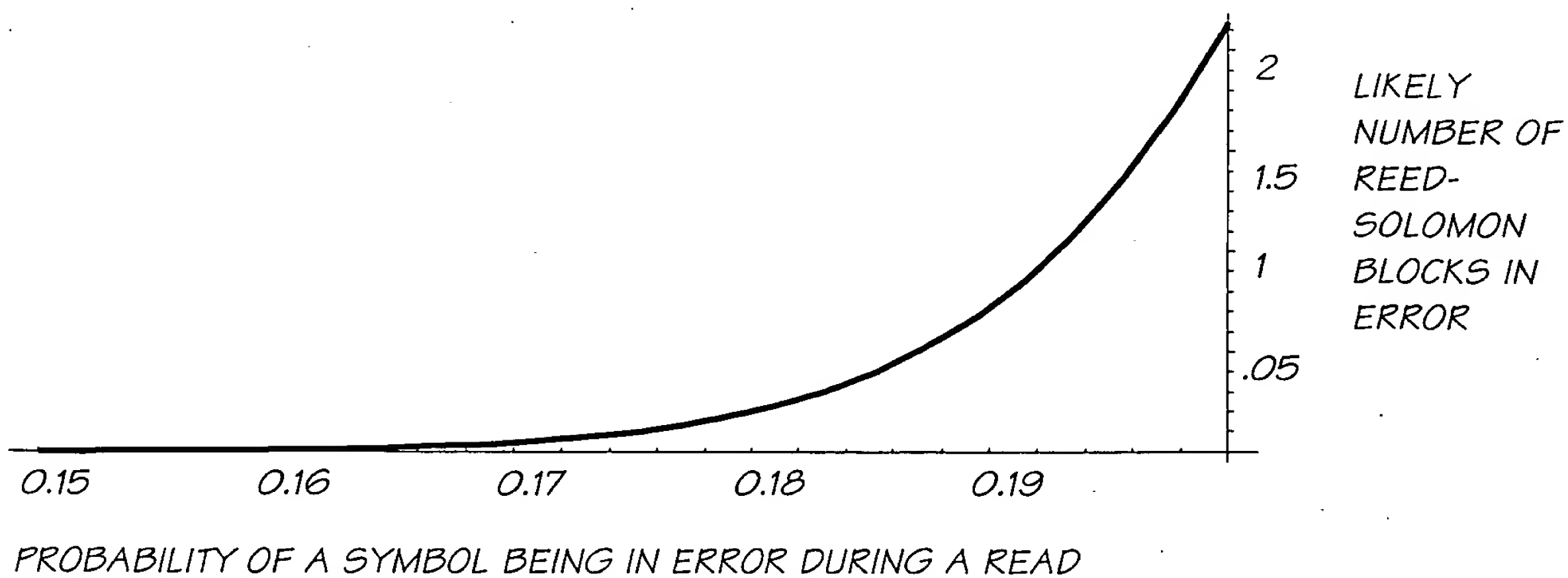
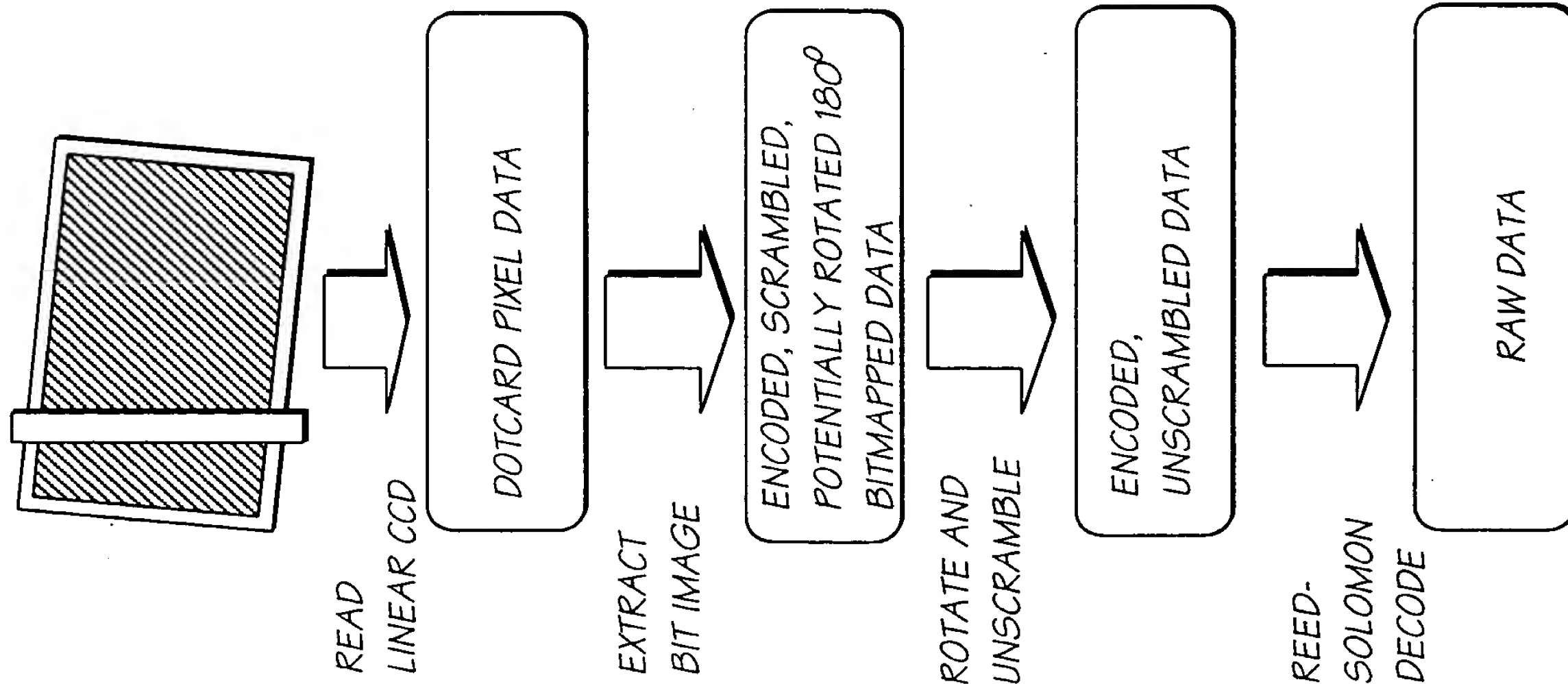


FIG. 66



APPROXIMATE DATA SIZES FOR 1600 DPI DOTCARD

86MM + 1MM IN HORIZONTAL DIMENSION FOR 0° ROTATION = 87MM

87MM = 16,252 SCANLINES

16,440 SCANLINES @ 11,000 PIXELS PER SCANLINE = 180,840,000 PIXELS

180,840,000 PIXELS @ 1 BYTE PER PIXEL = 180,840,000 BYTES = 172.5 MB

64 DATA BLOCKS, EACH CONTAINING 597 COLUMNS (595 DATA REGION COLUMNS AND 2 ORIENTATION COLUMNS), @ 48 BYTES PER COLUMN = 28,656 BYTES PER DATA BLOCK FOR A TOTAL OF 1,833,984 BYTES.

64 DATA BLOCKS, EACH CONTAINING 112 ENCODED REED SOLOMON BLOCKS, @ 255 BYTES PER REED SOLOMON BLOCK FOR A TOTAL OF 1,827,840 BYTES.

DECODED DATA, WITH A MAXIMUM SIZE OF 910,082 BYTES.
(64 X 112 X 127 - (2 CONTROL BLOCKS @ 127 BYTES))

FIG. 67

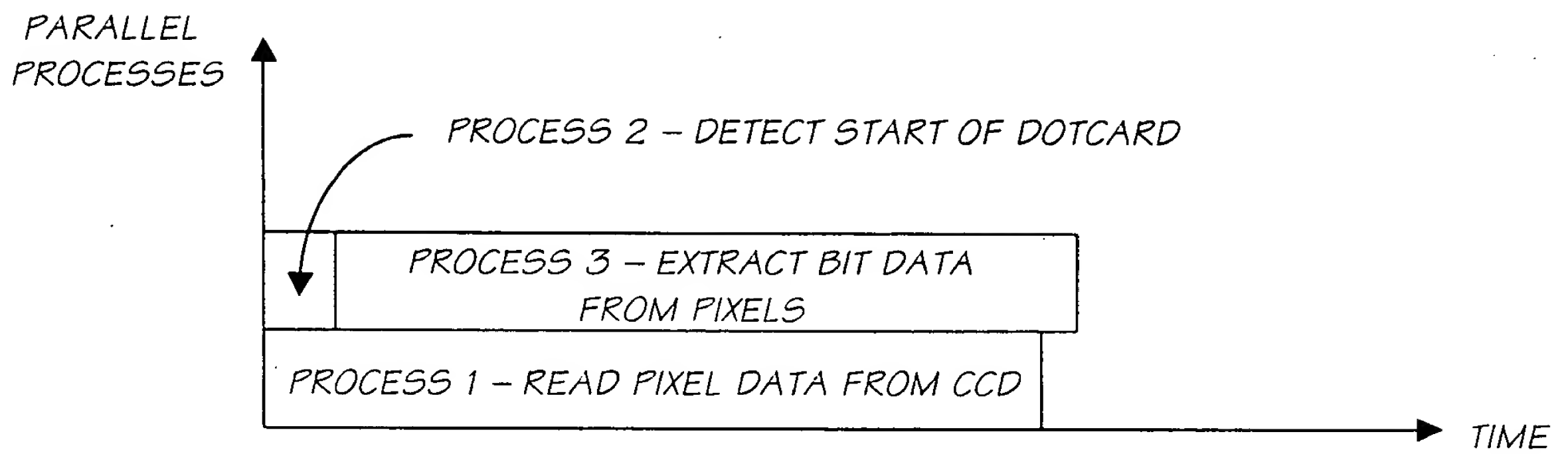


FIG. 68

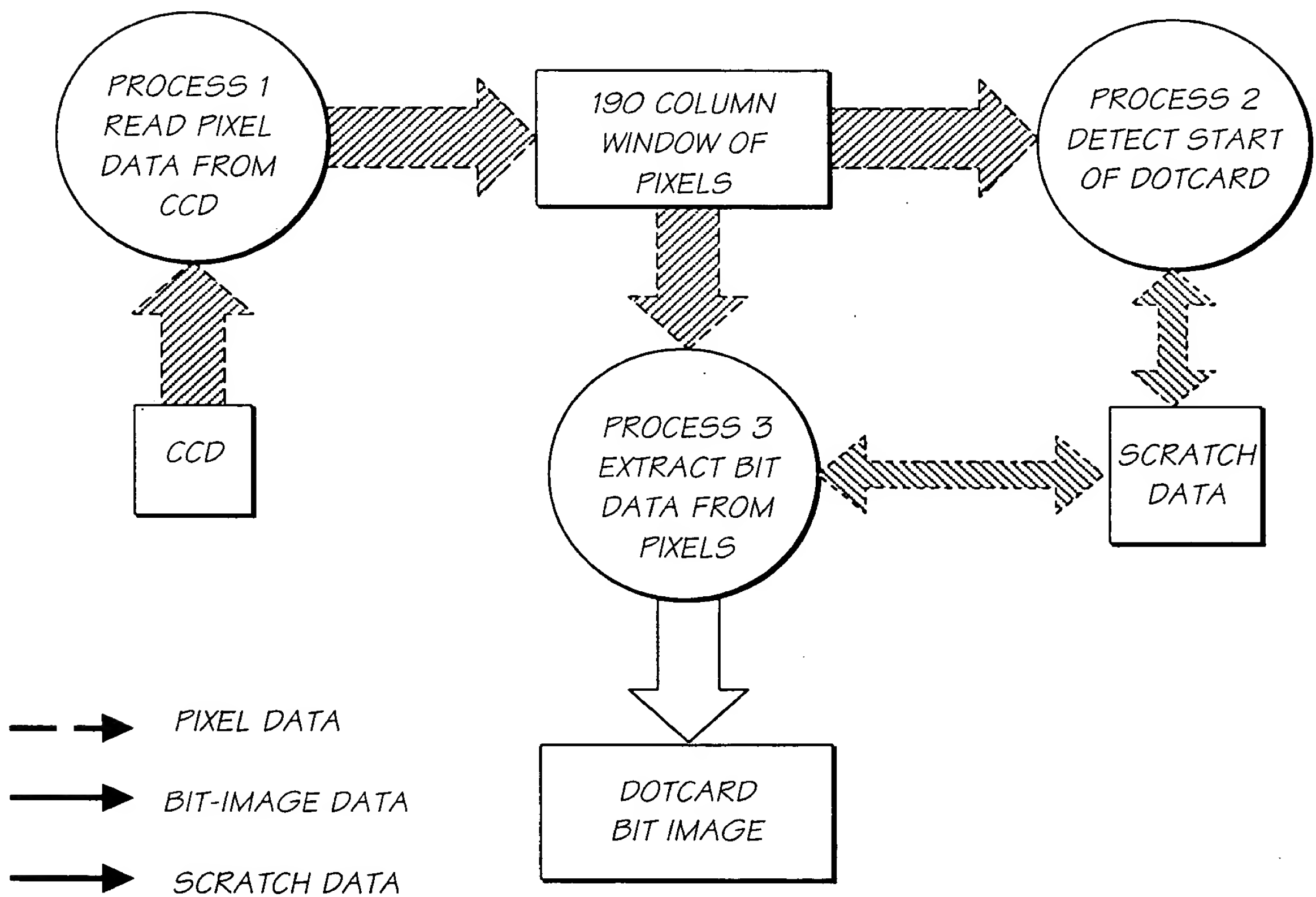


FIG. 69

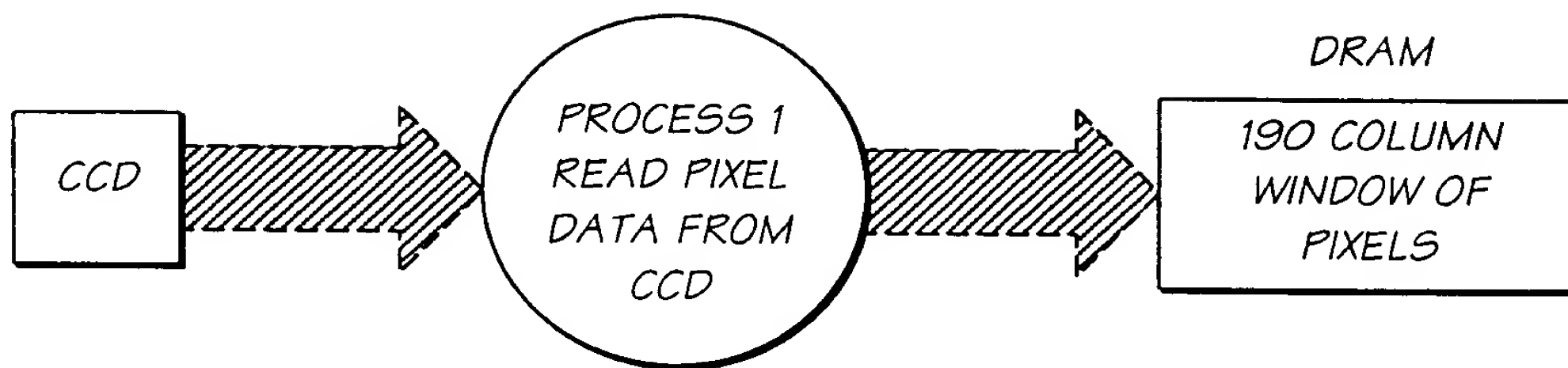


FIG. 70

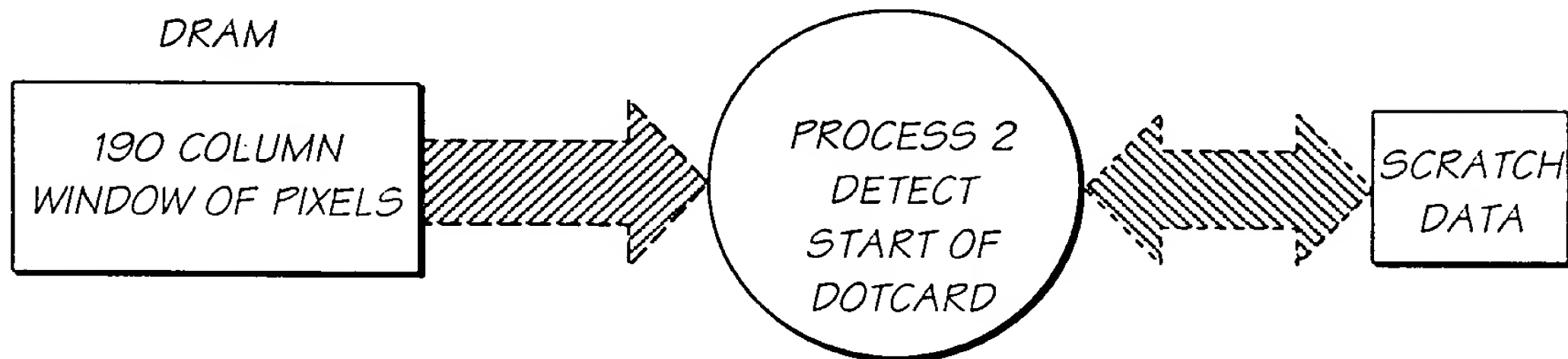


FIG. 71

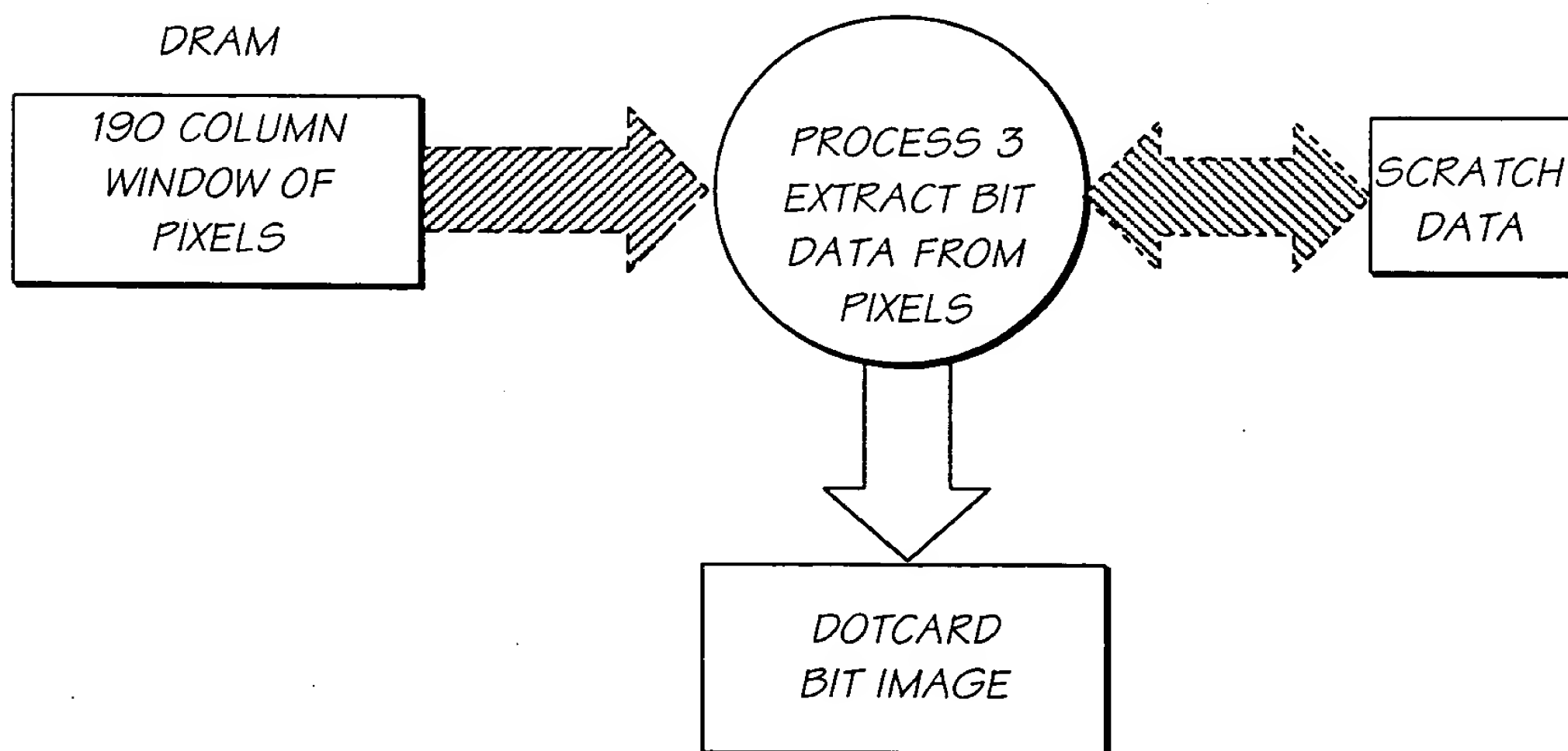


FIG. 72

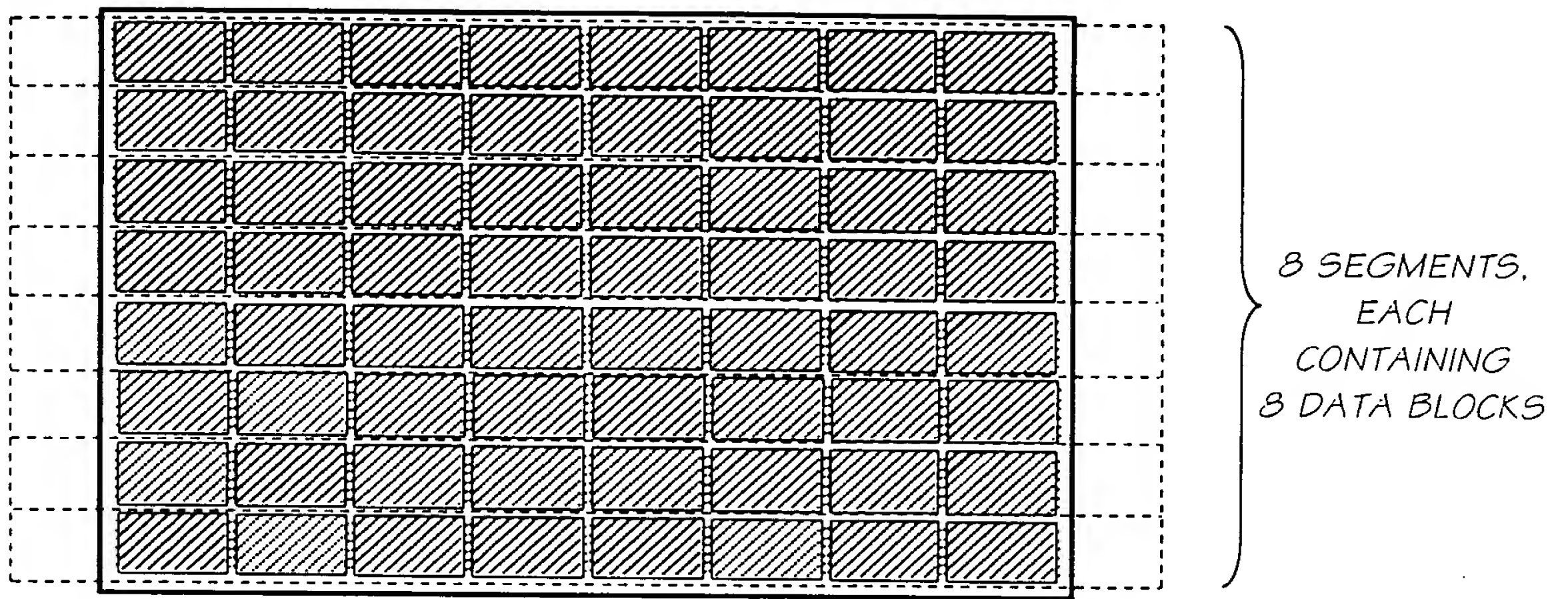


FIG. 73

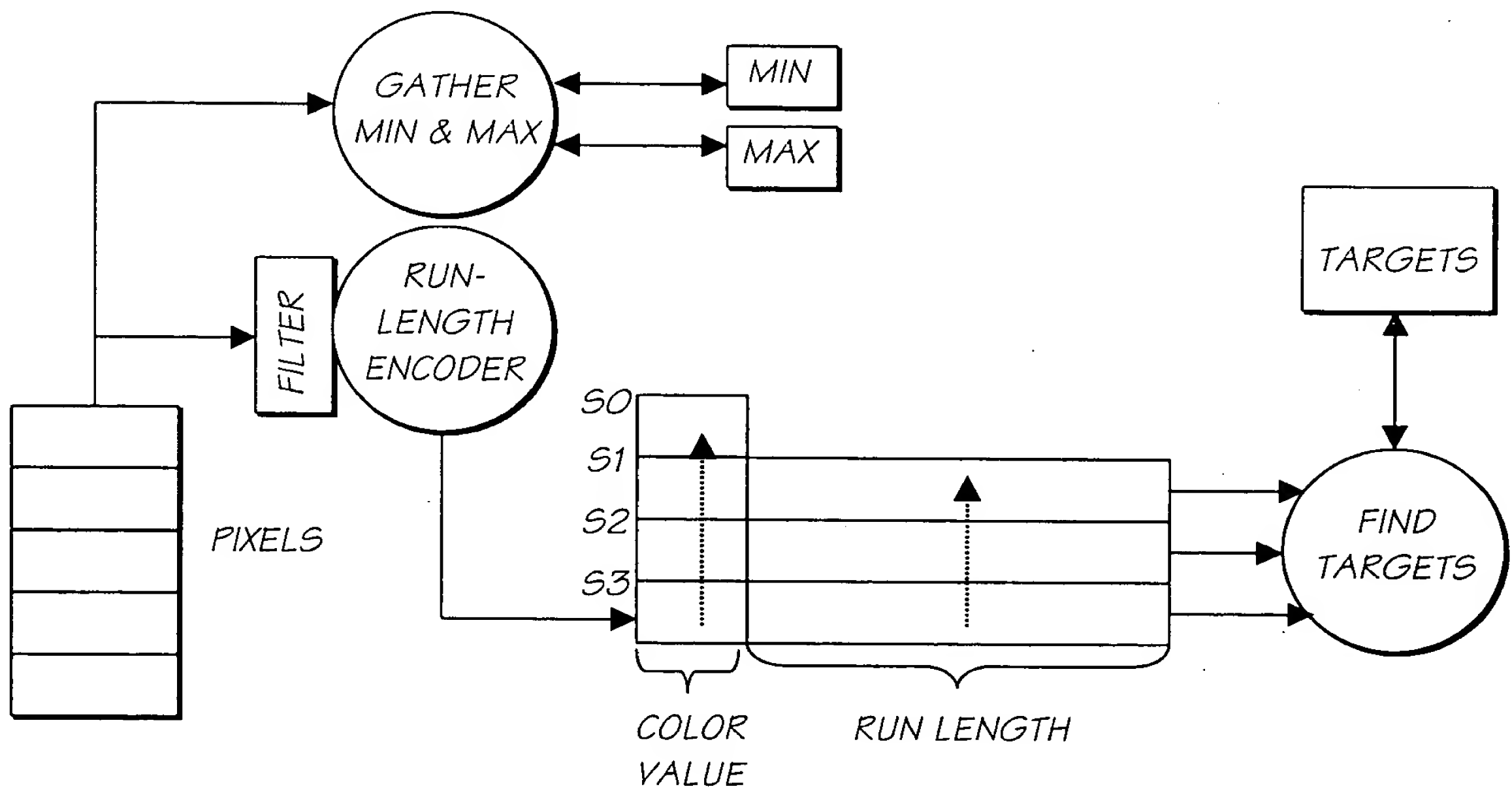


FIG. 74

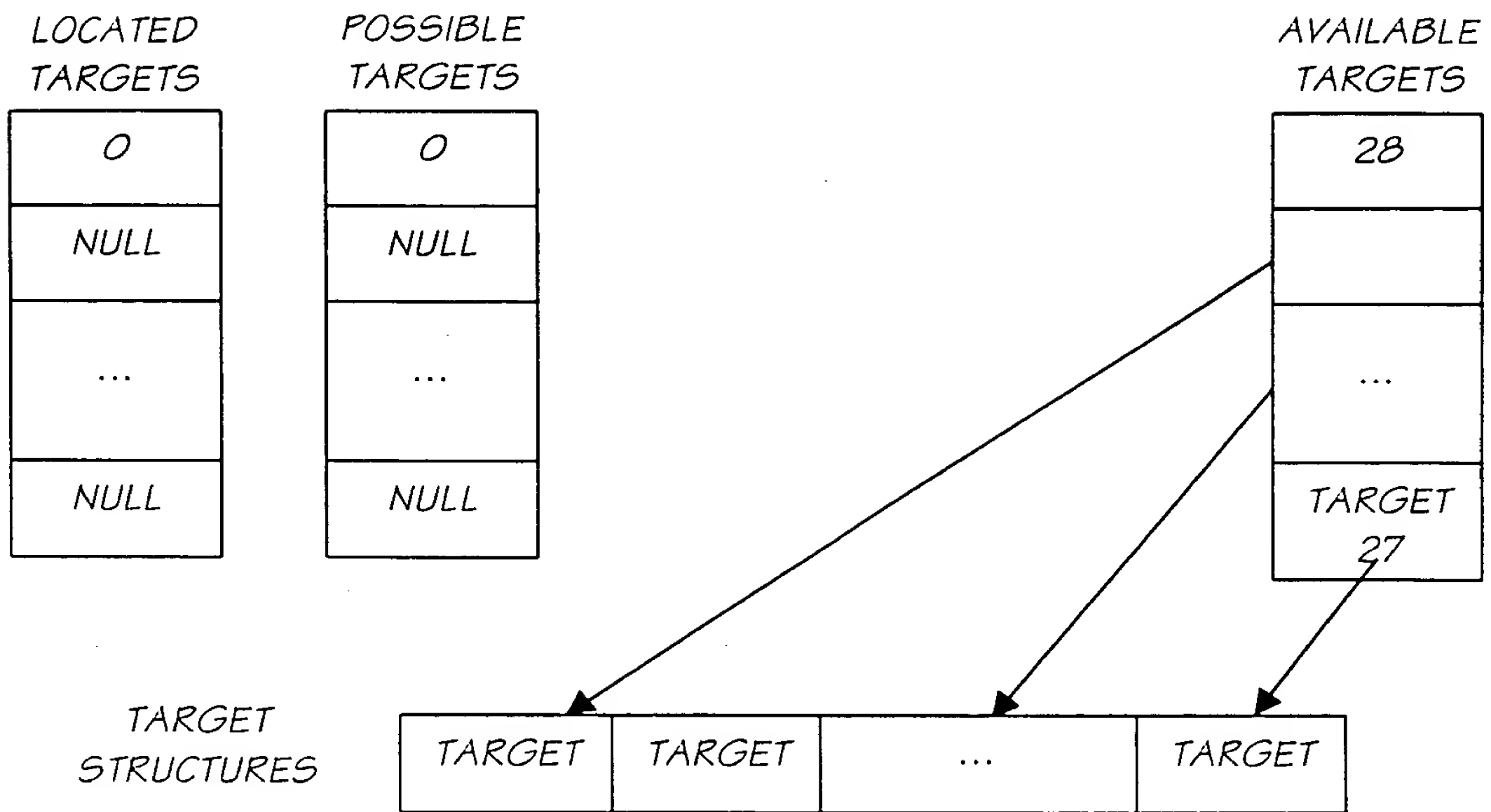


FIG. 75

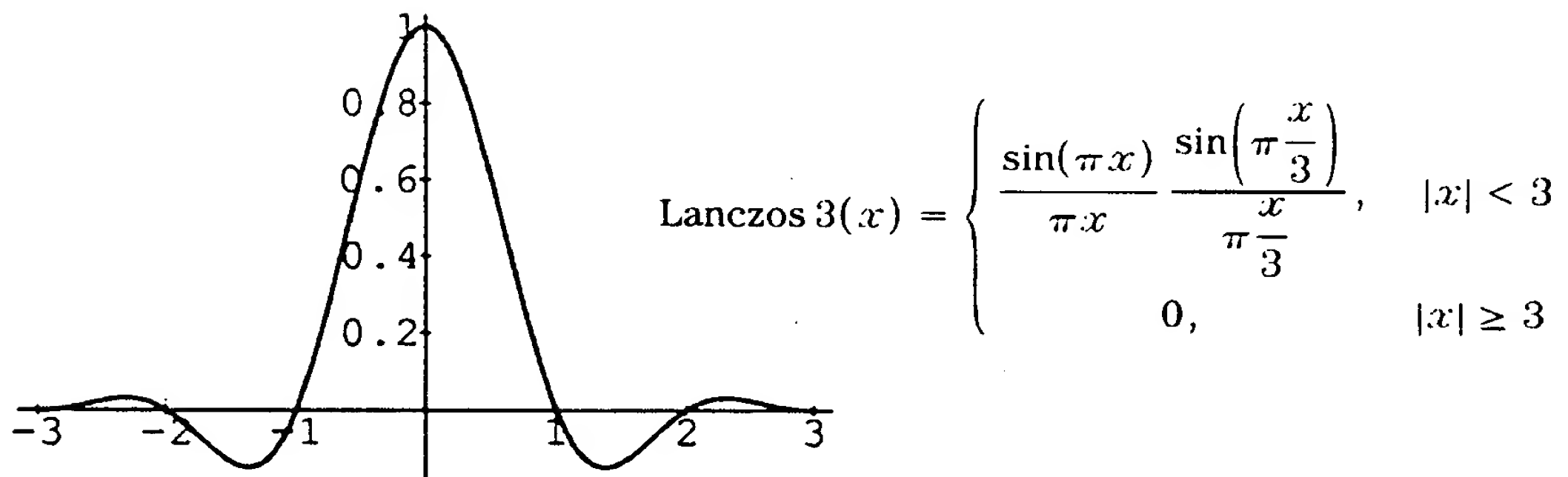


FIG. 76

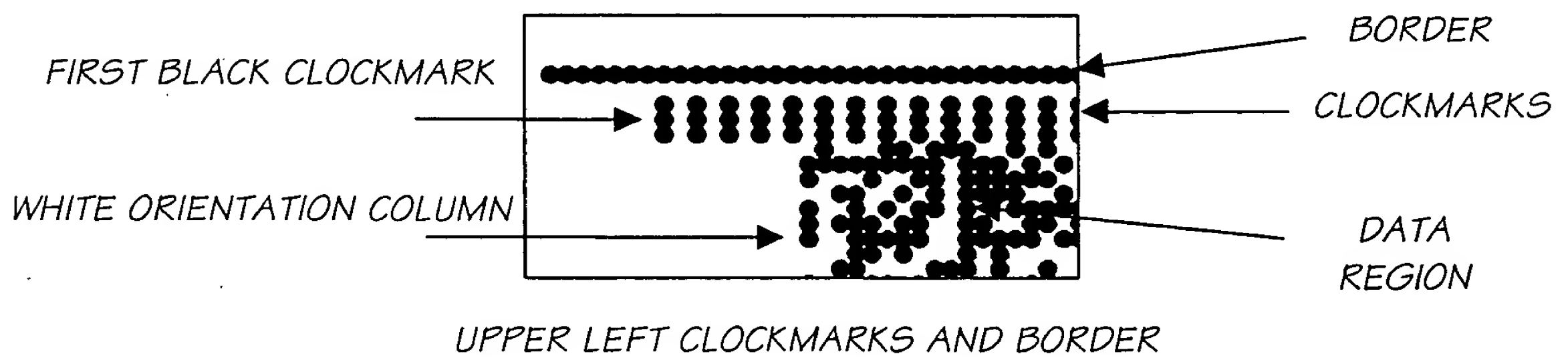


FIG. 77

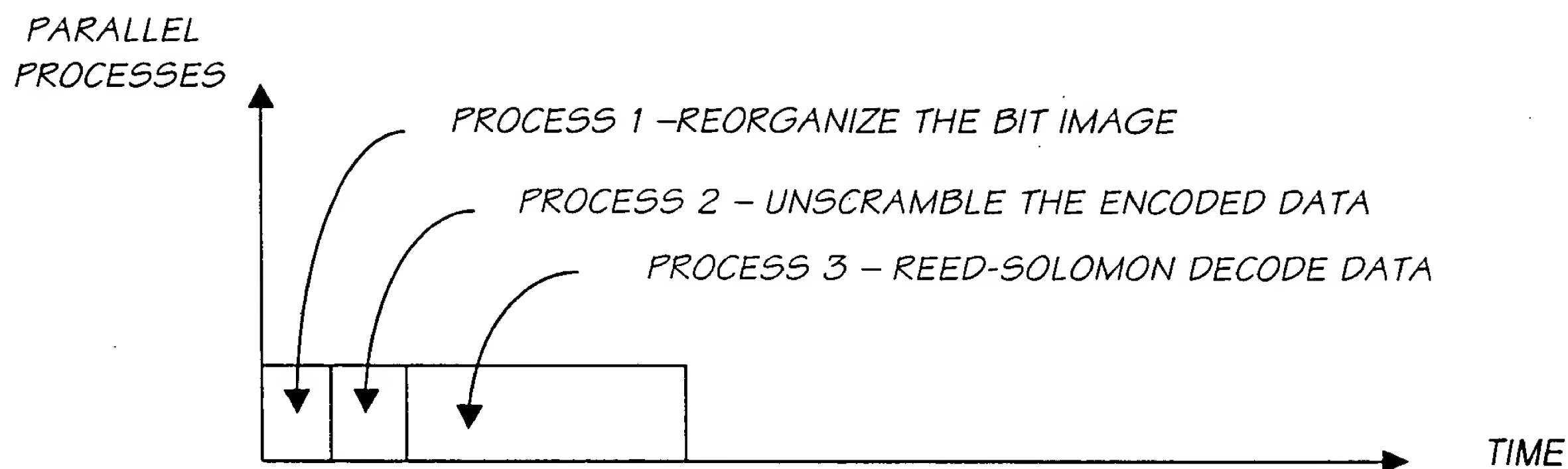


FIG. 78

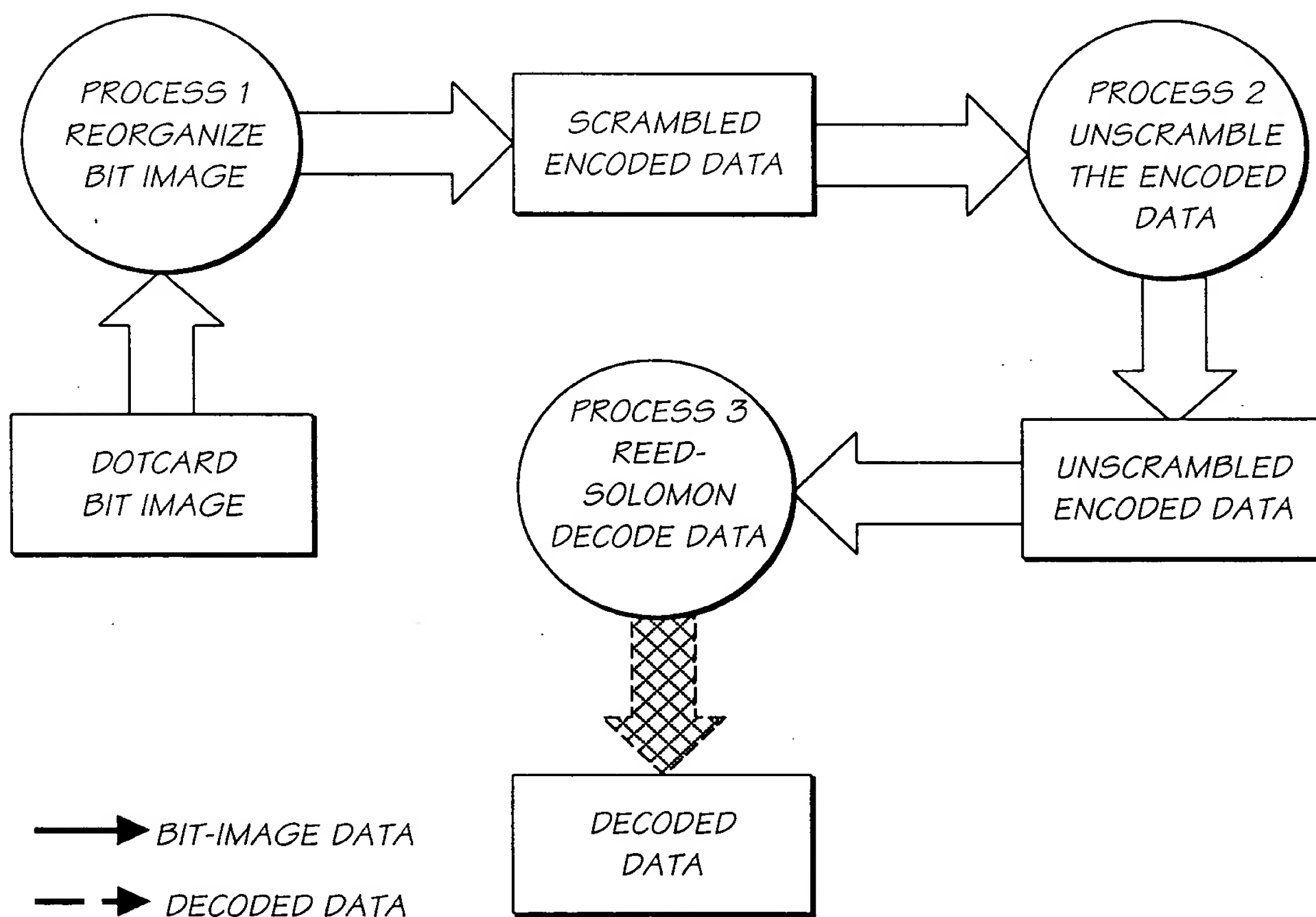


FIG. 79

FIG. 80

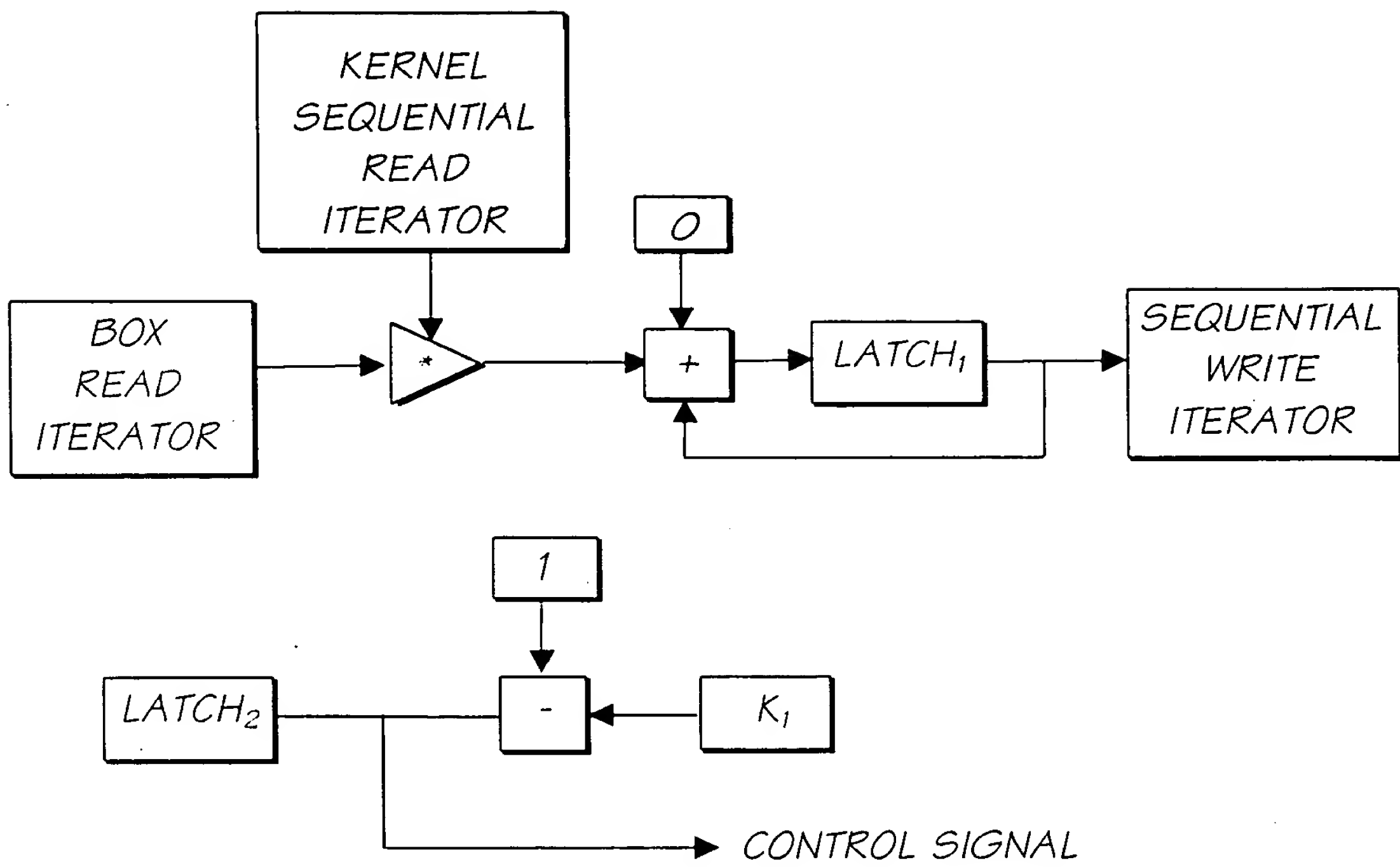


FIG. 81

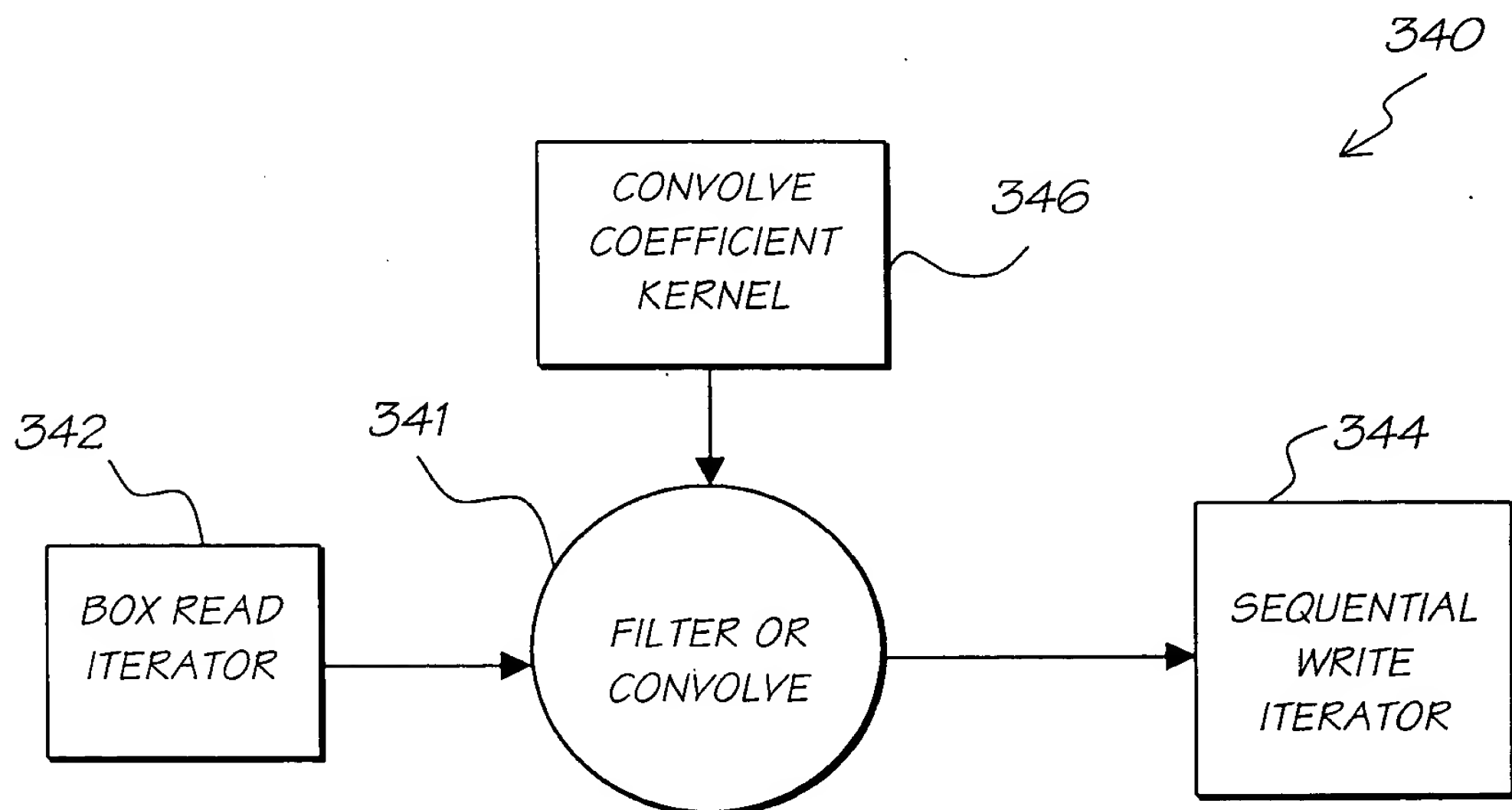


FIG. 82

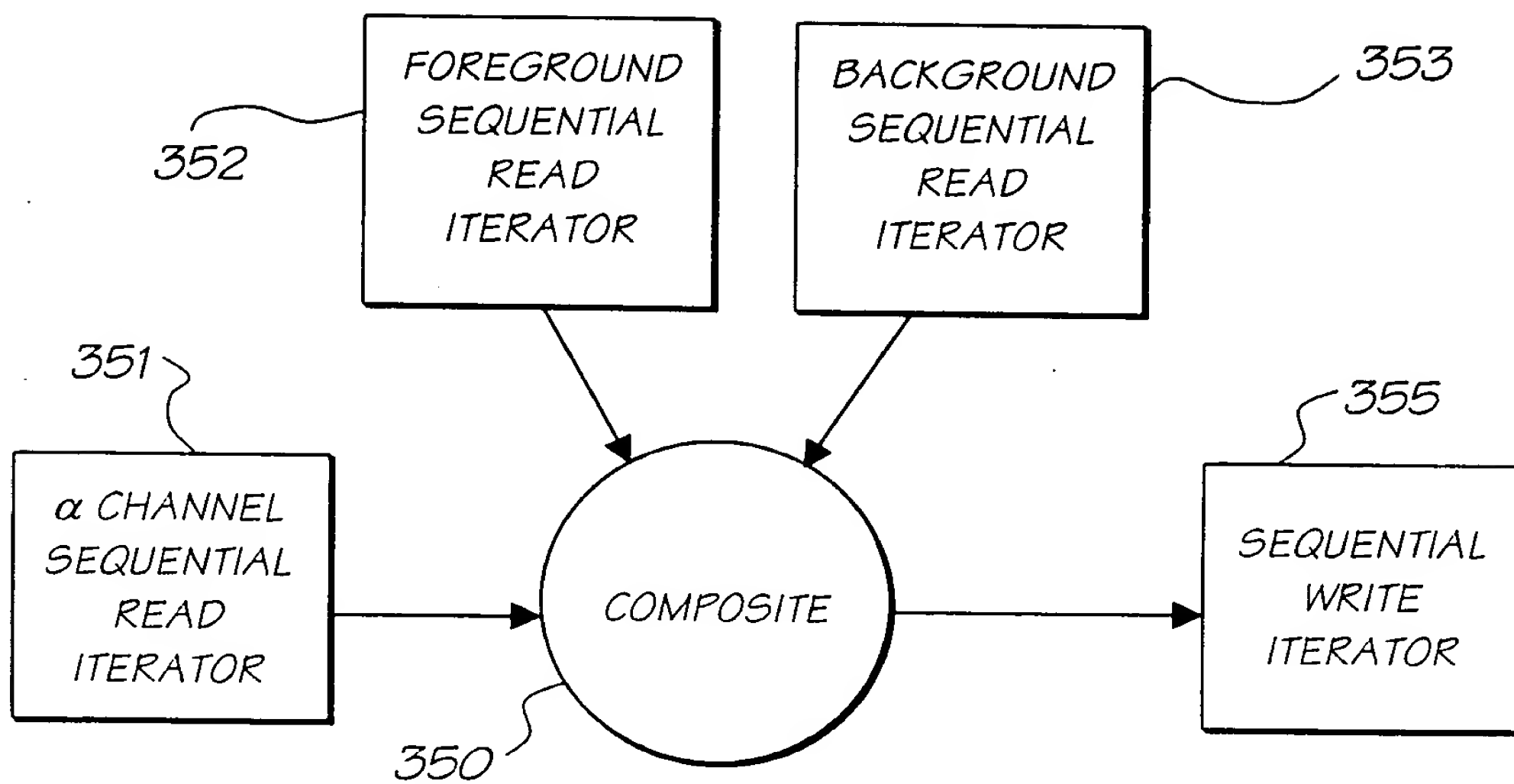


FIG. 83

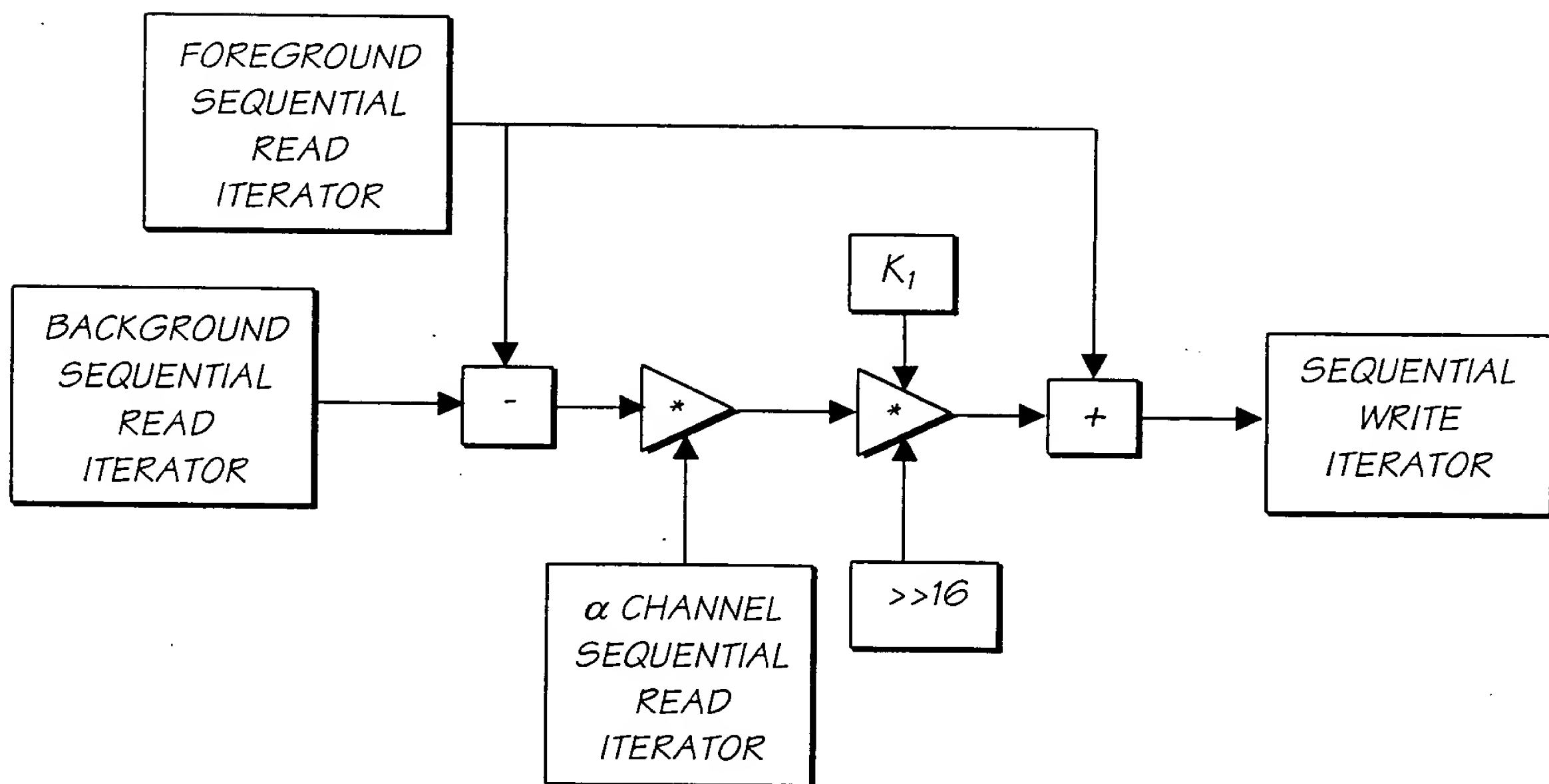


FIG. 84

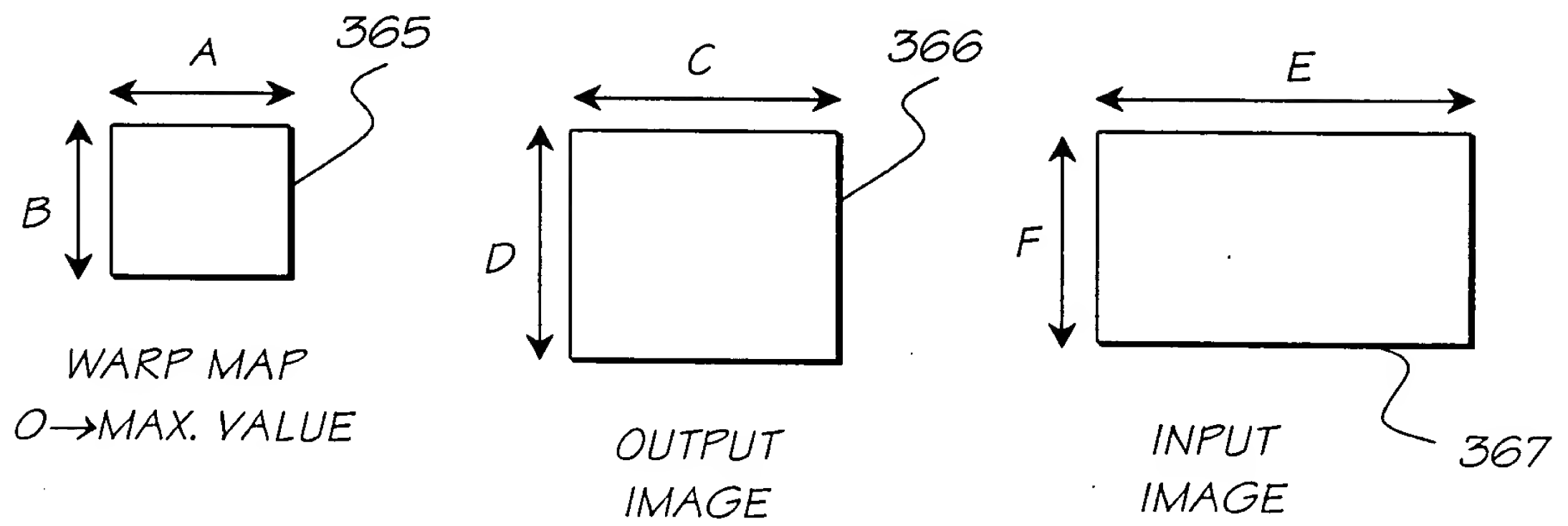


FIG. 85

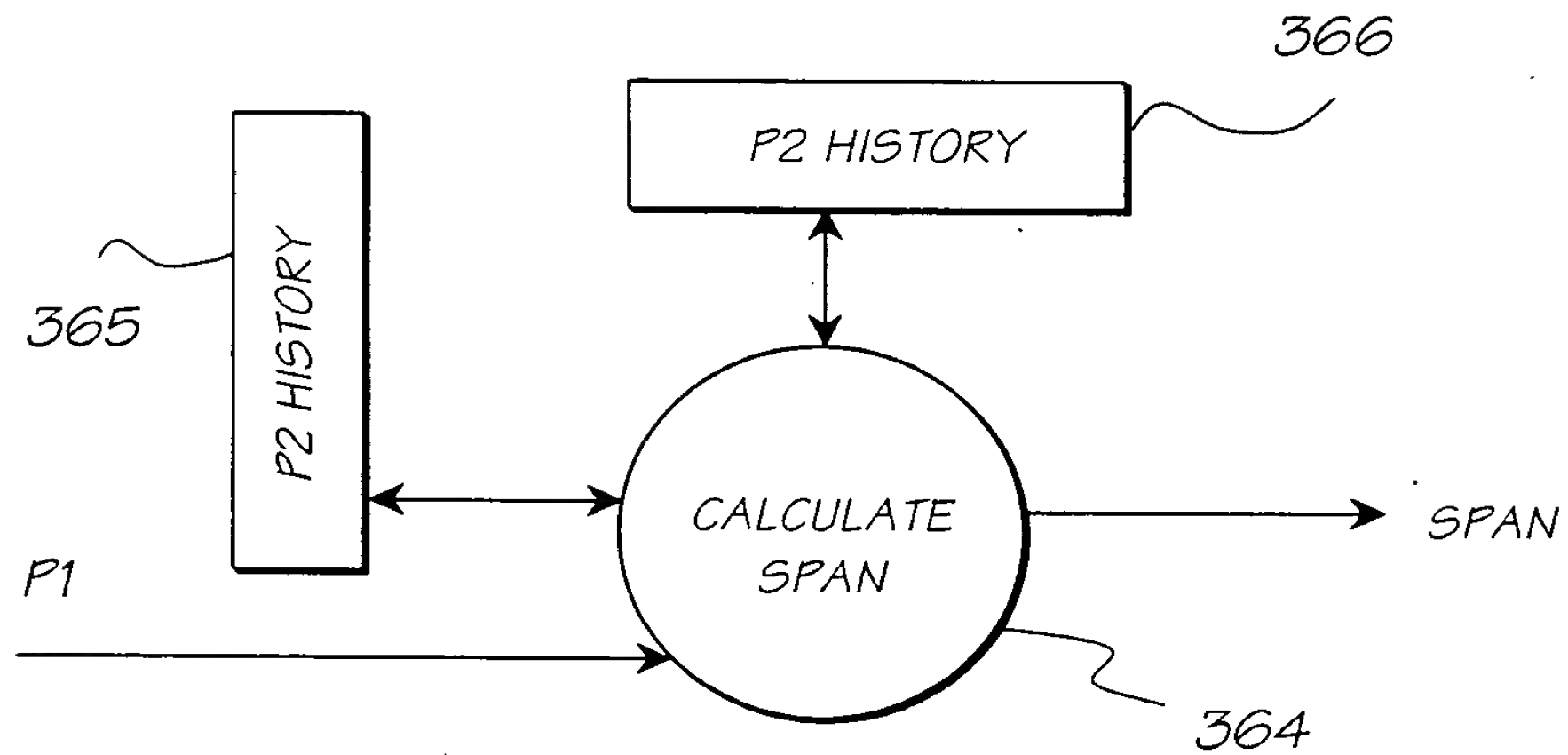


FIG. 86

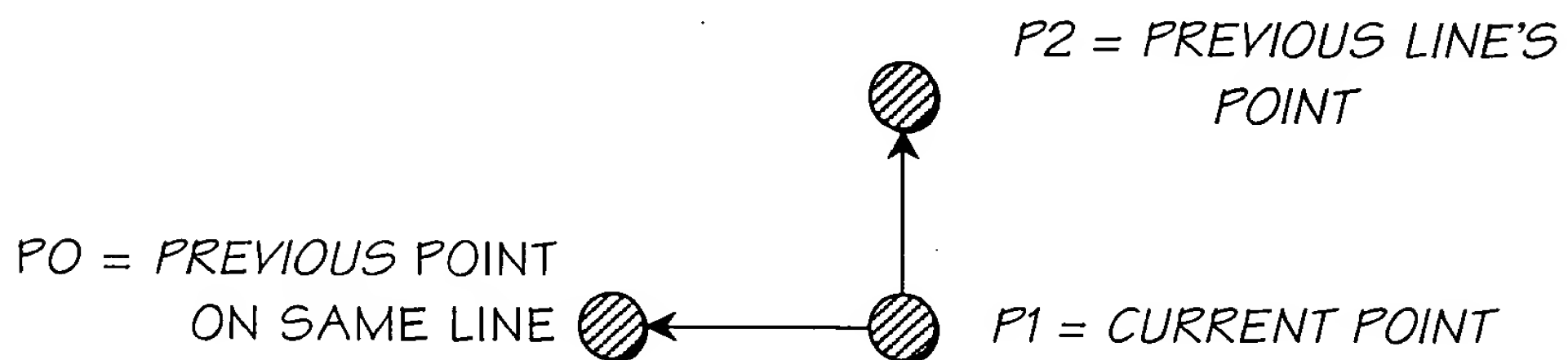


FIG. 88

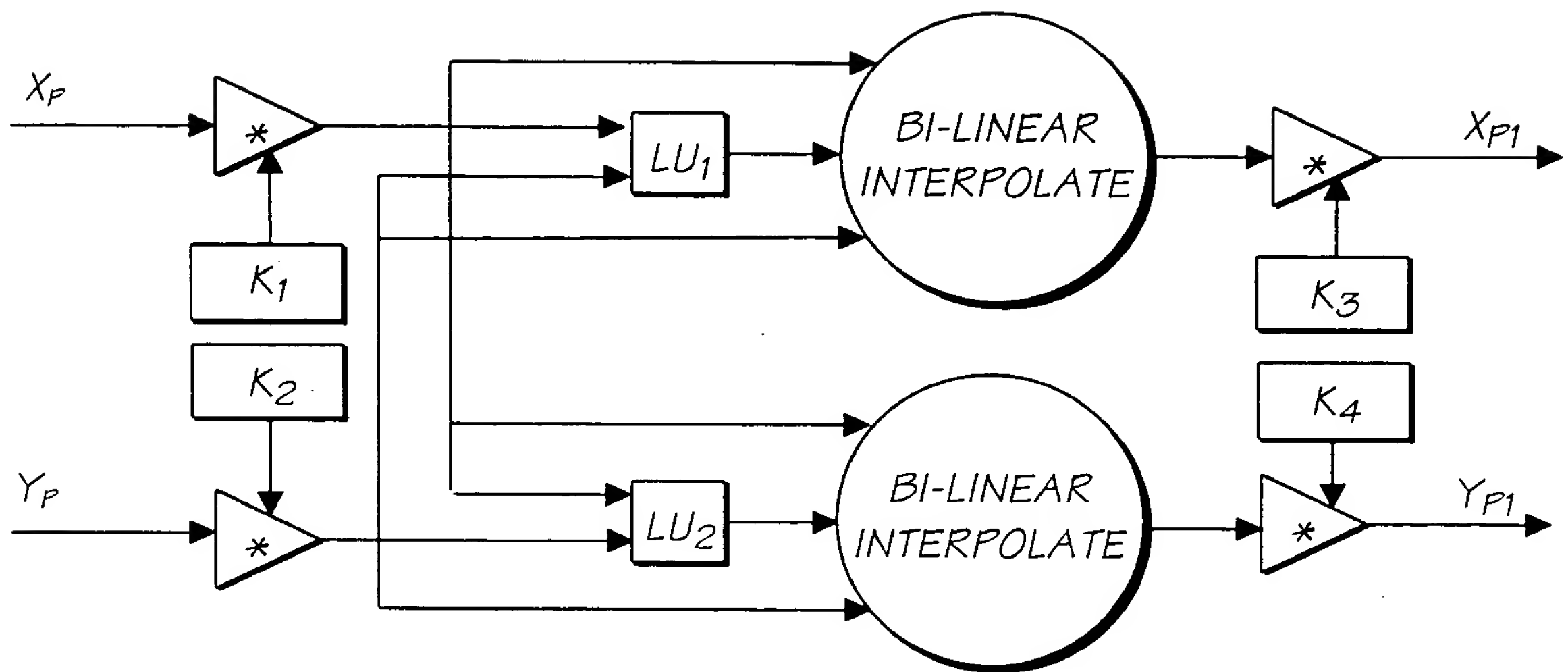


FIG. 87

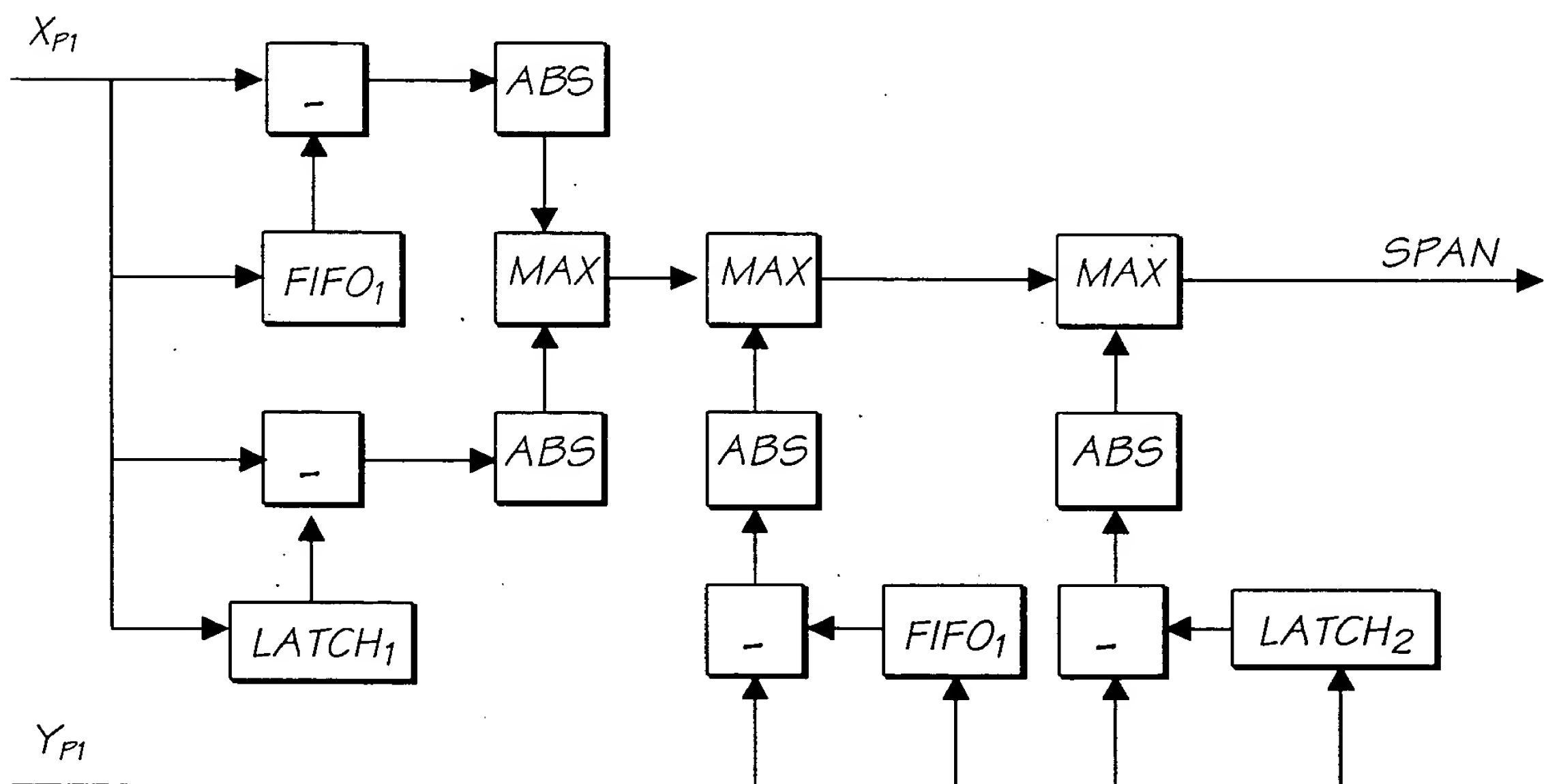


FIG. 89

POINT (x, y) ON LEVEL B
OF PYRAMID

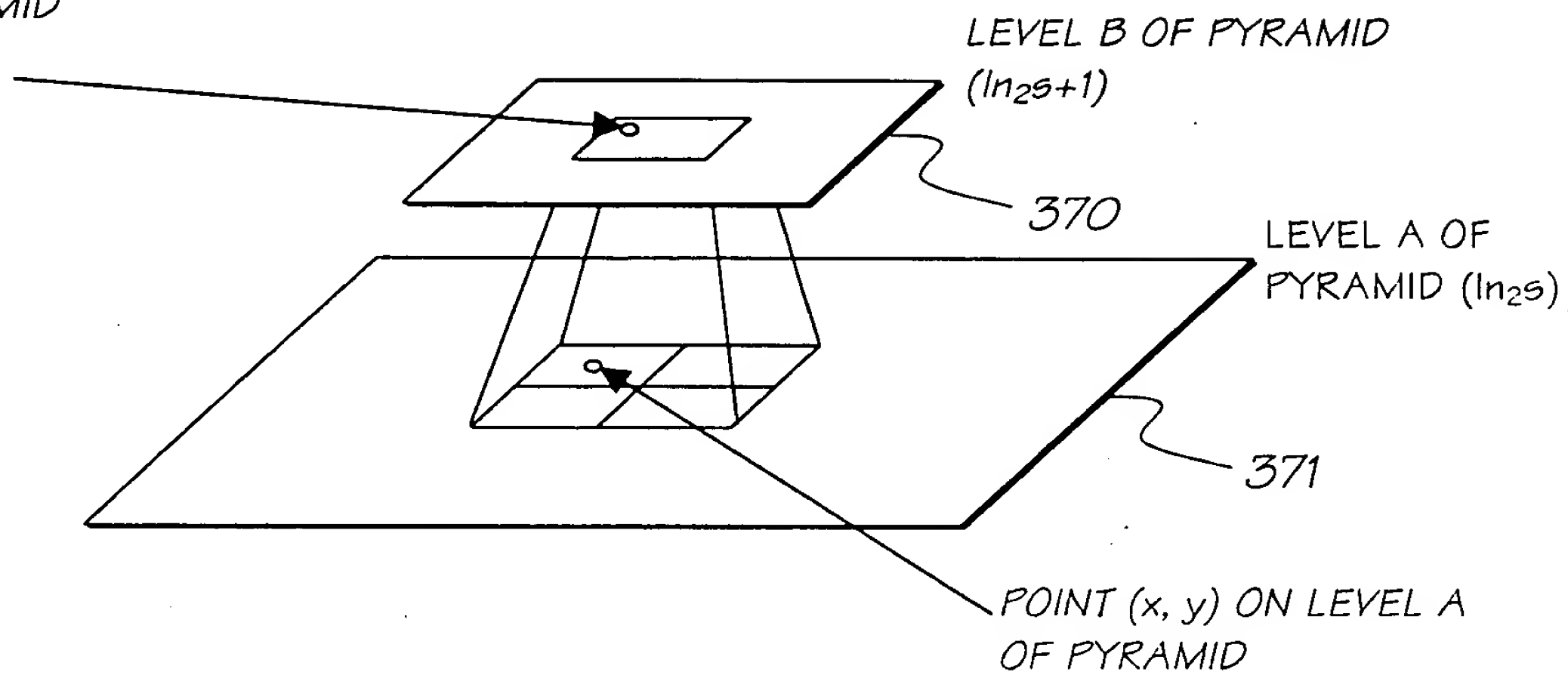


FIG. 90

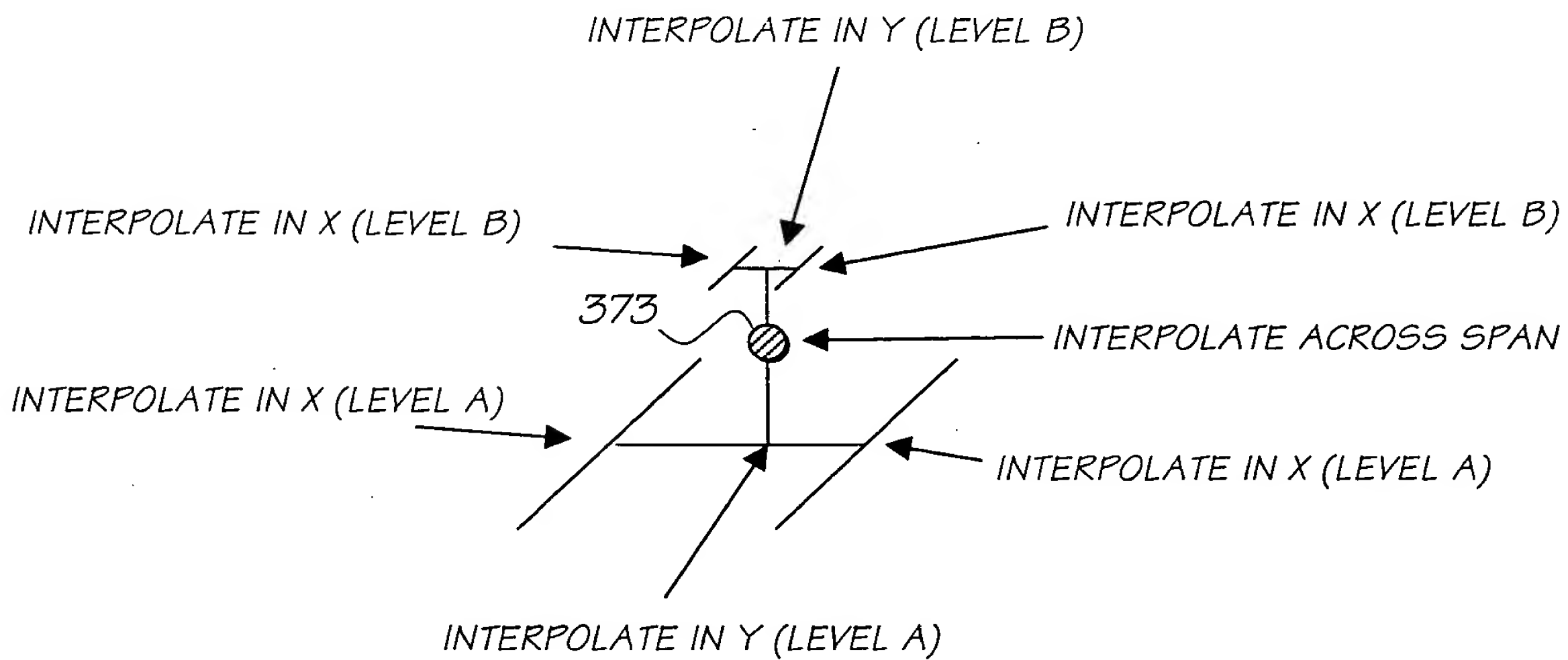


FIG. 91

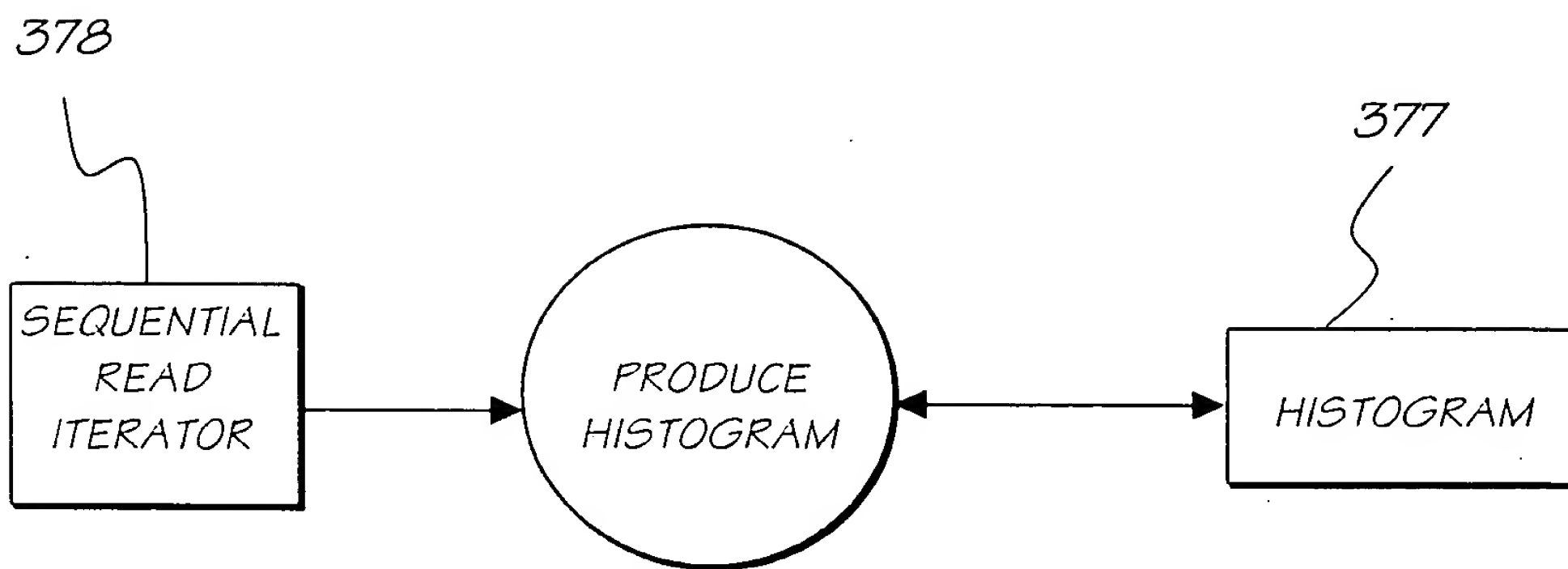


FIG. 92

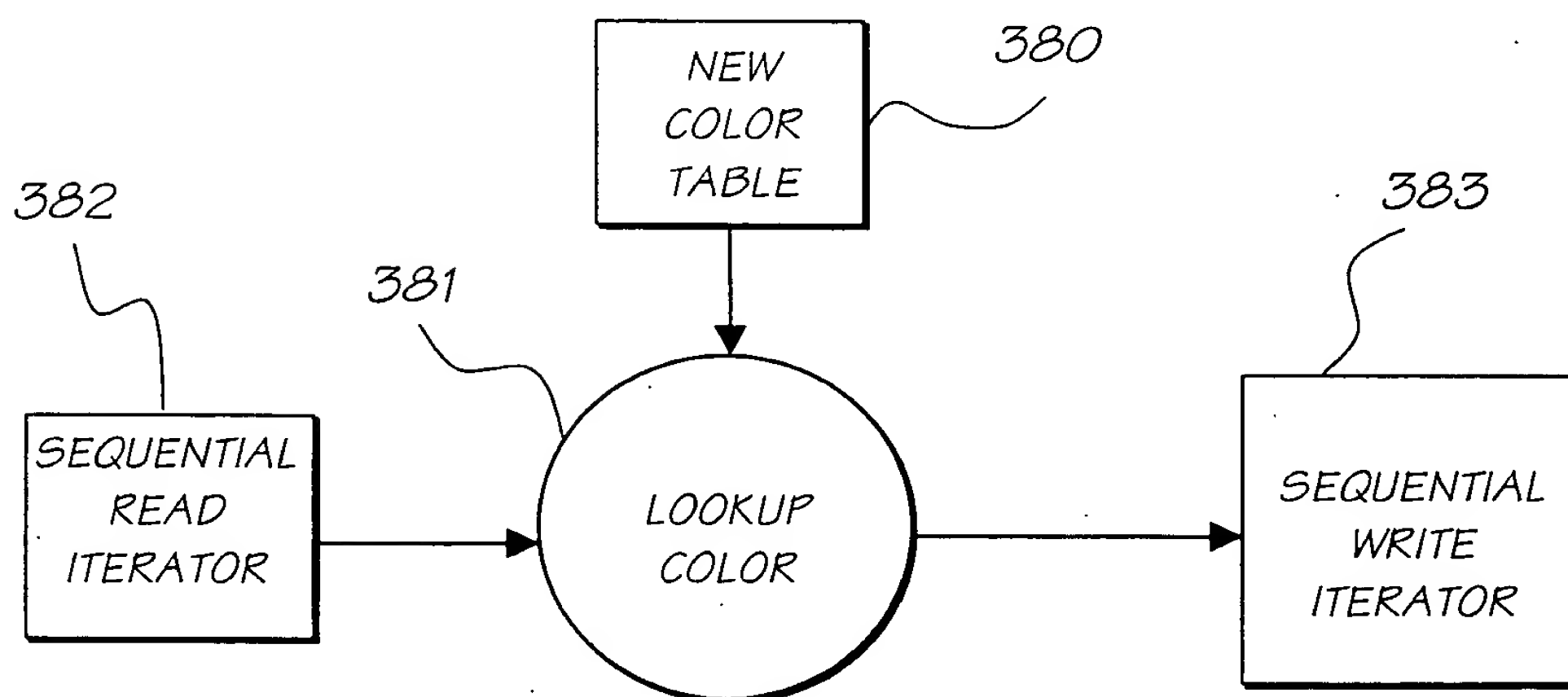


FIG. 93

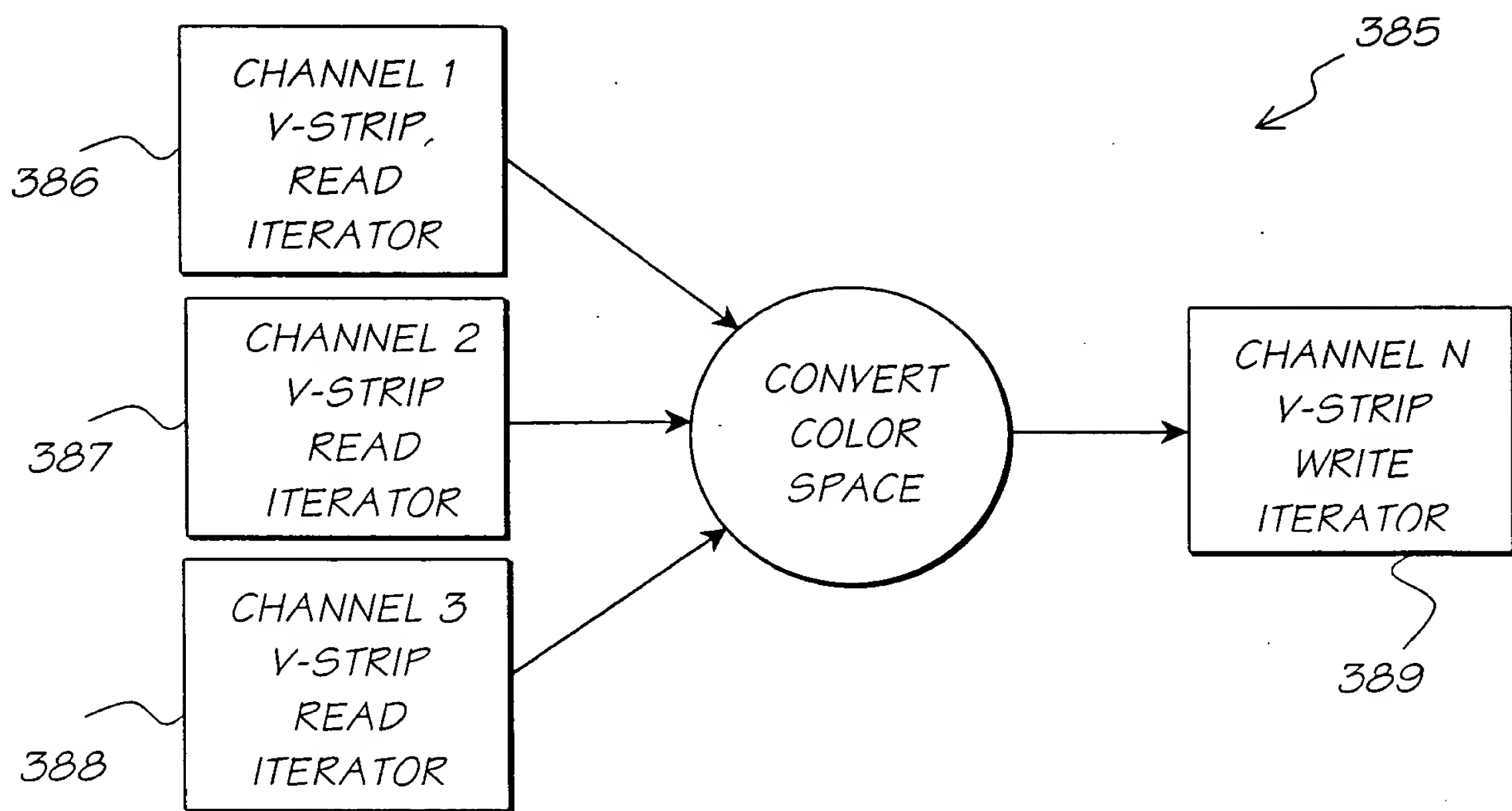


FIG. 94

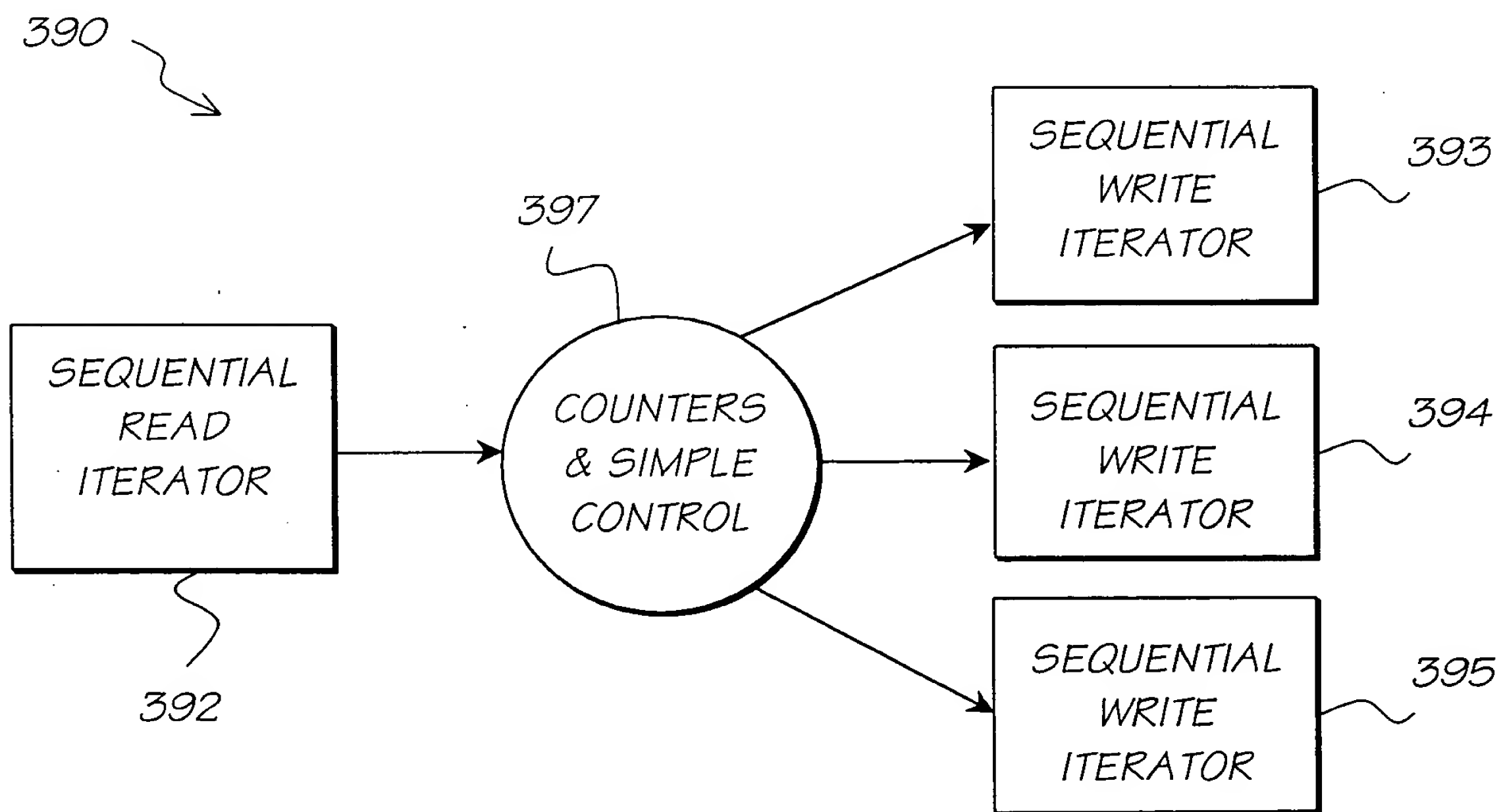


FIG. 101

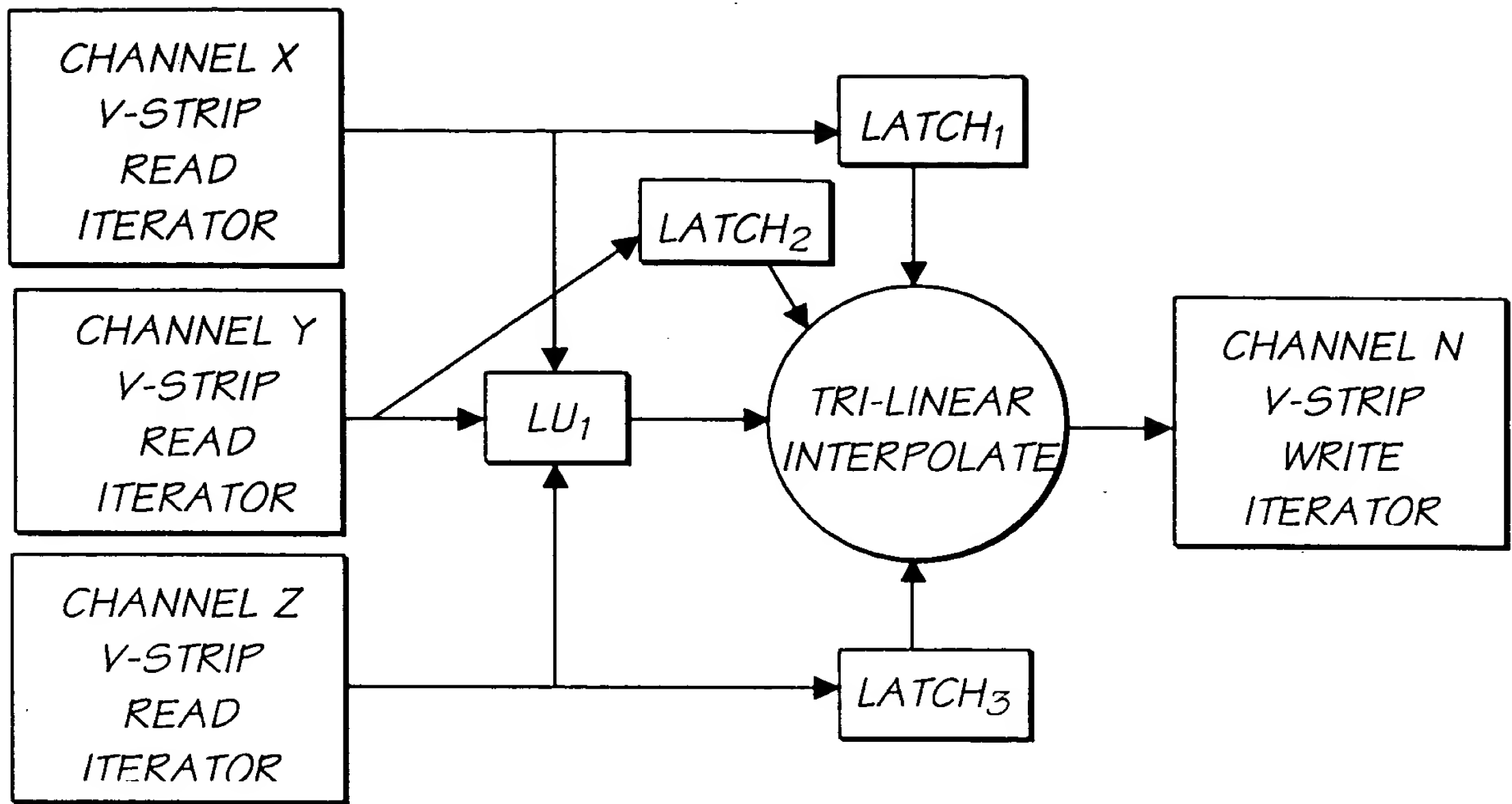


FIG. 95

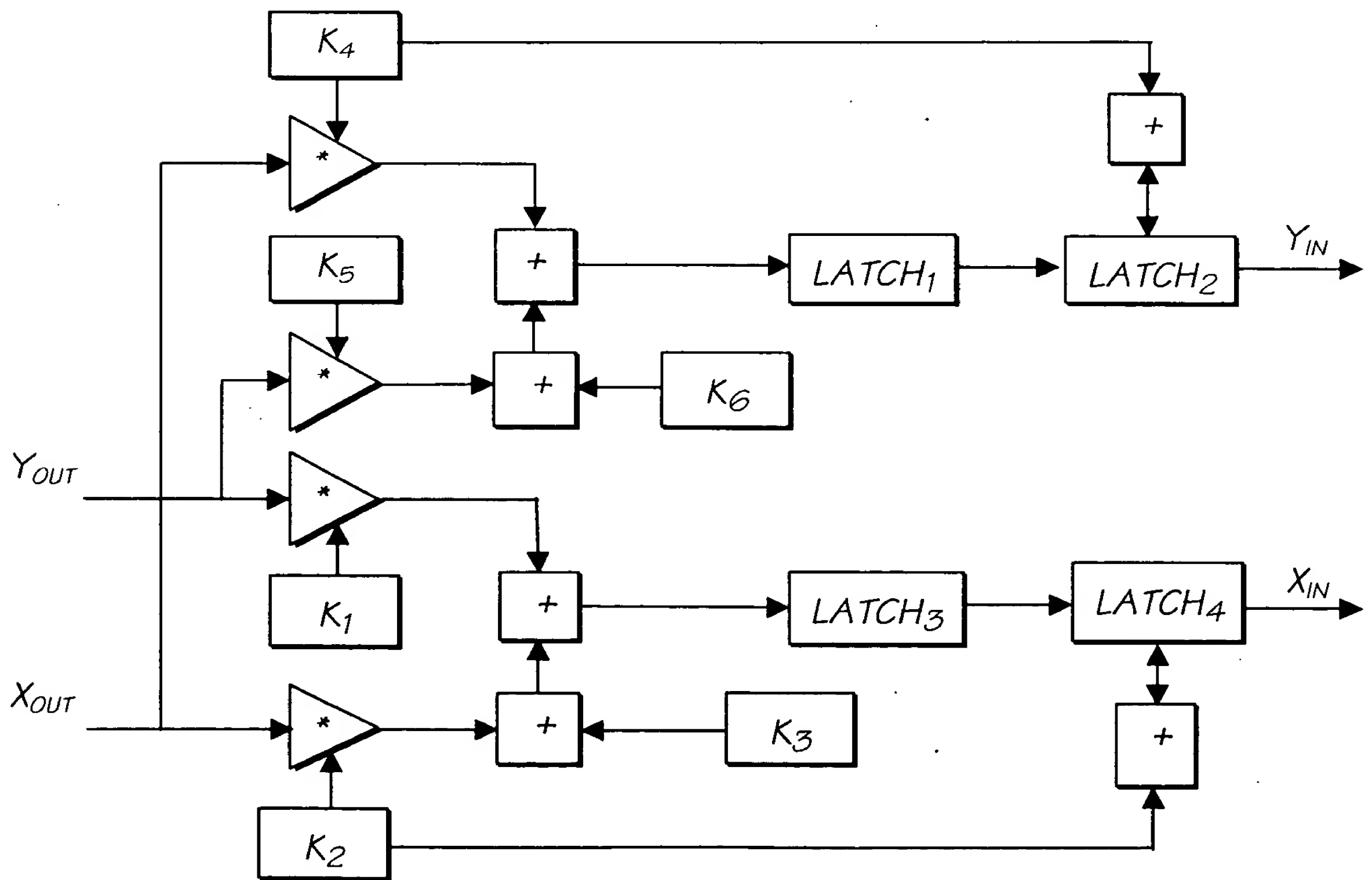


FIG. 96

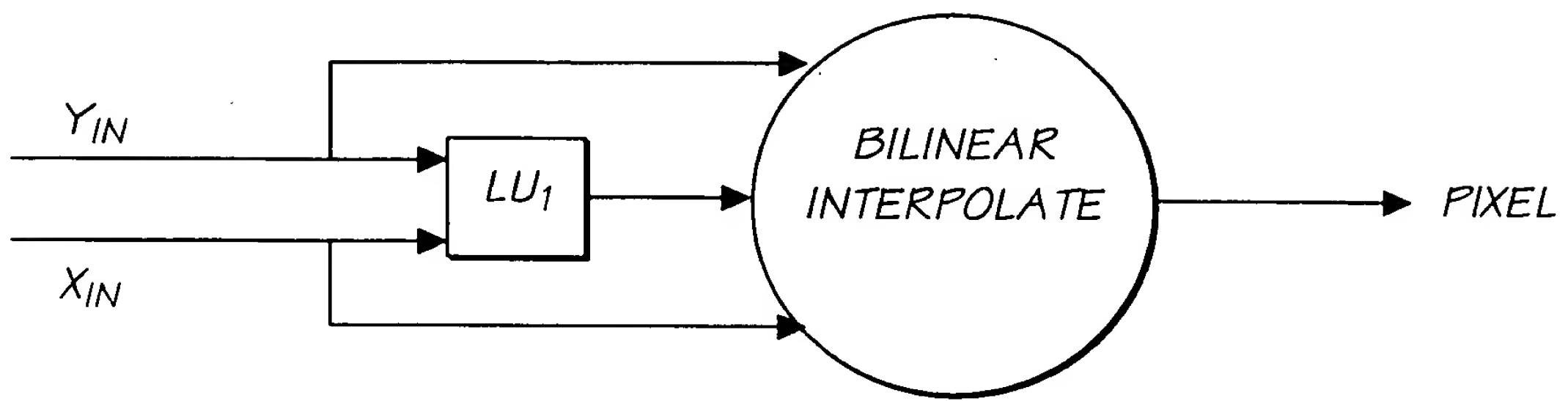


FIG. 97

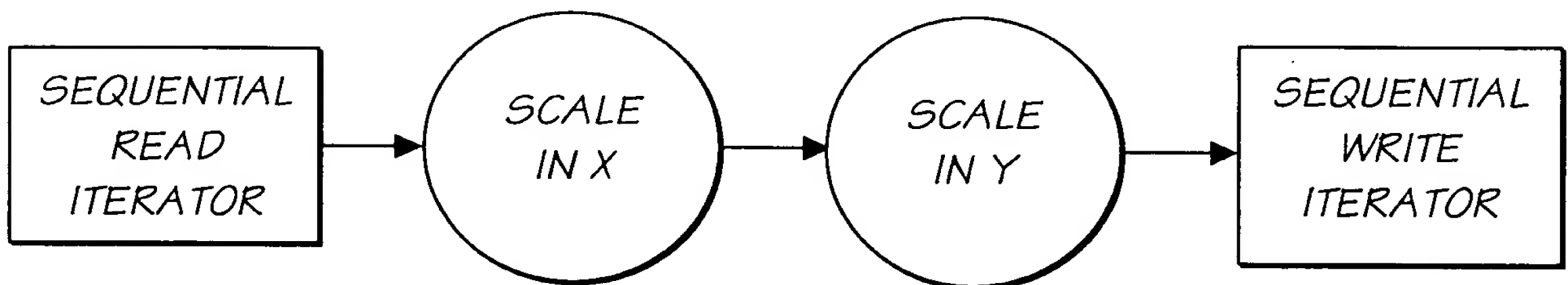


FIG. 98

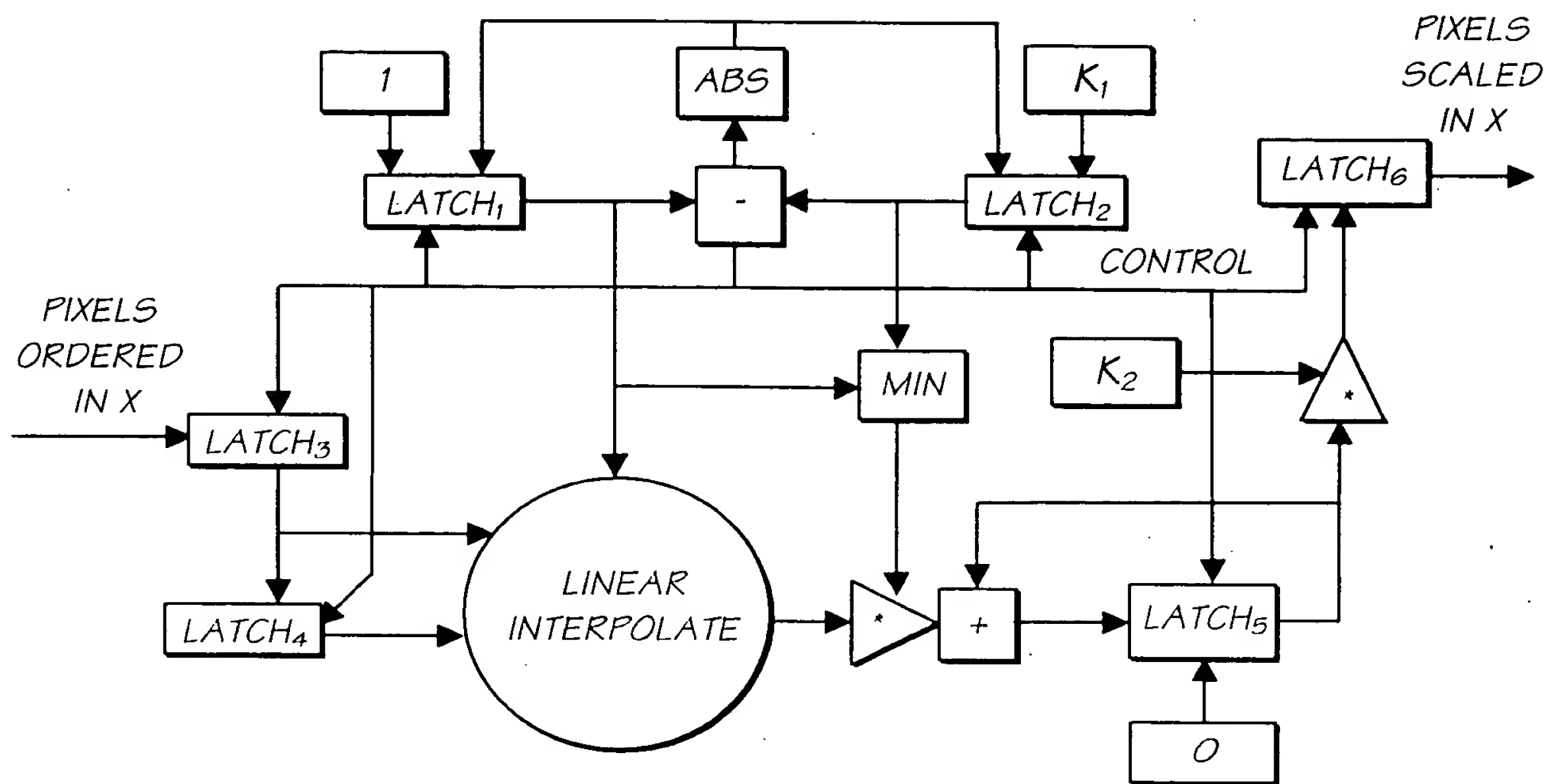


FIG. 99

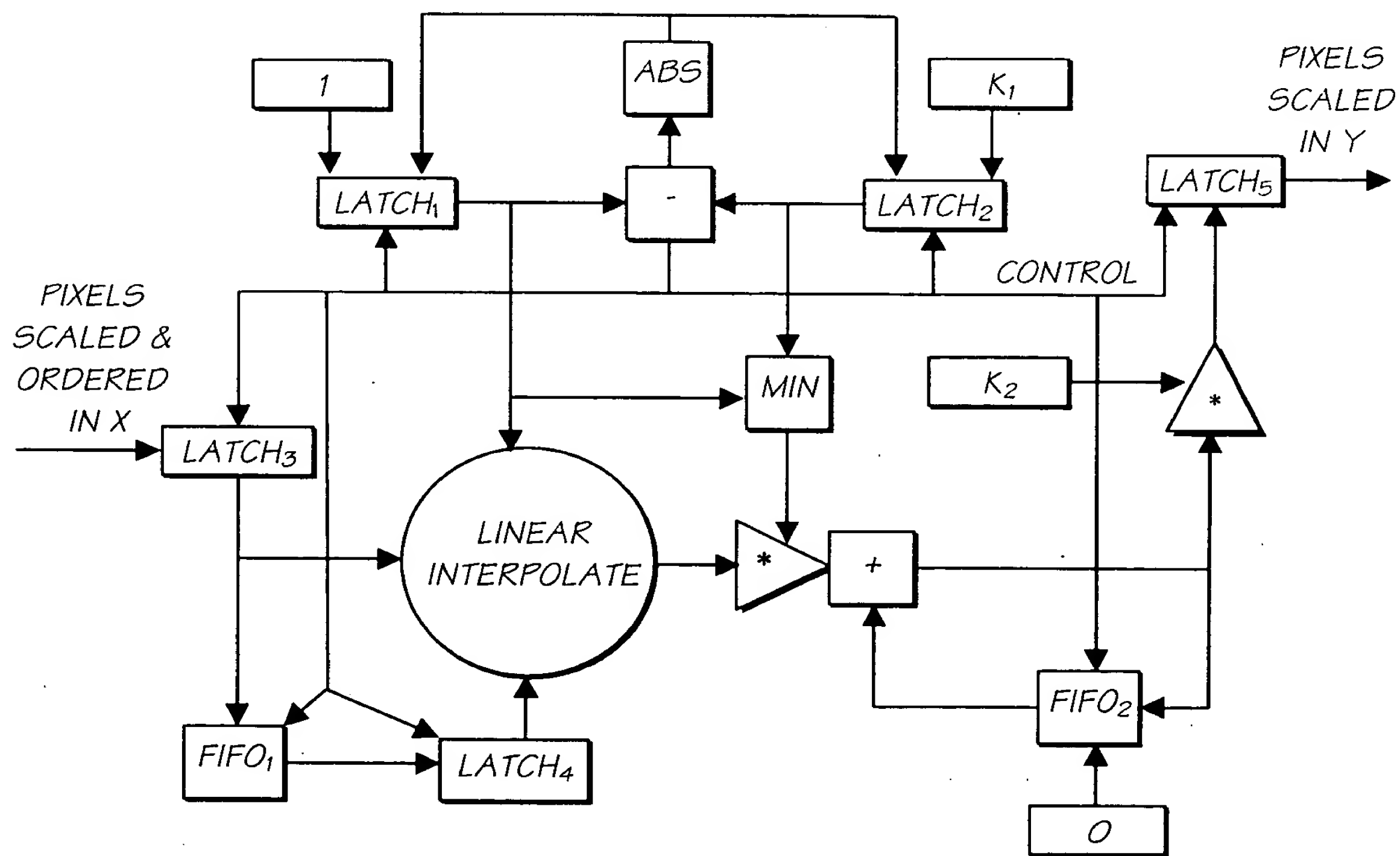


FIG. 100

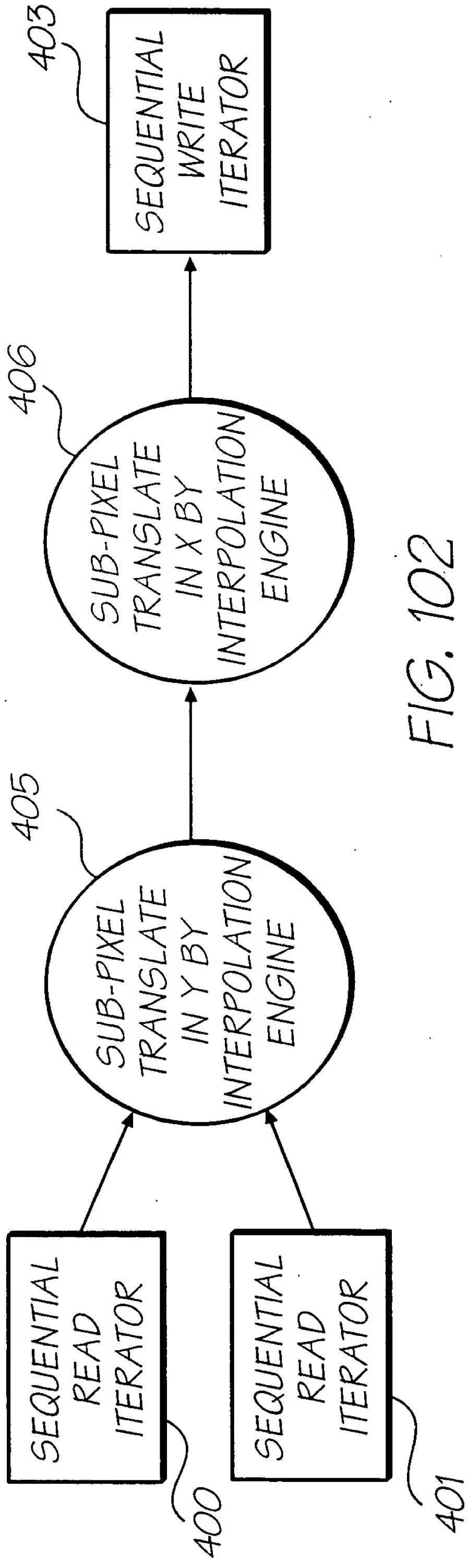


FIG. 102

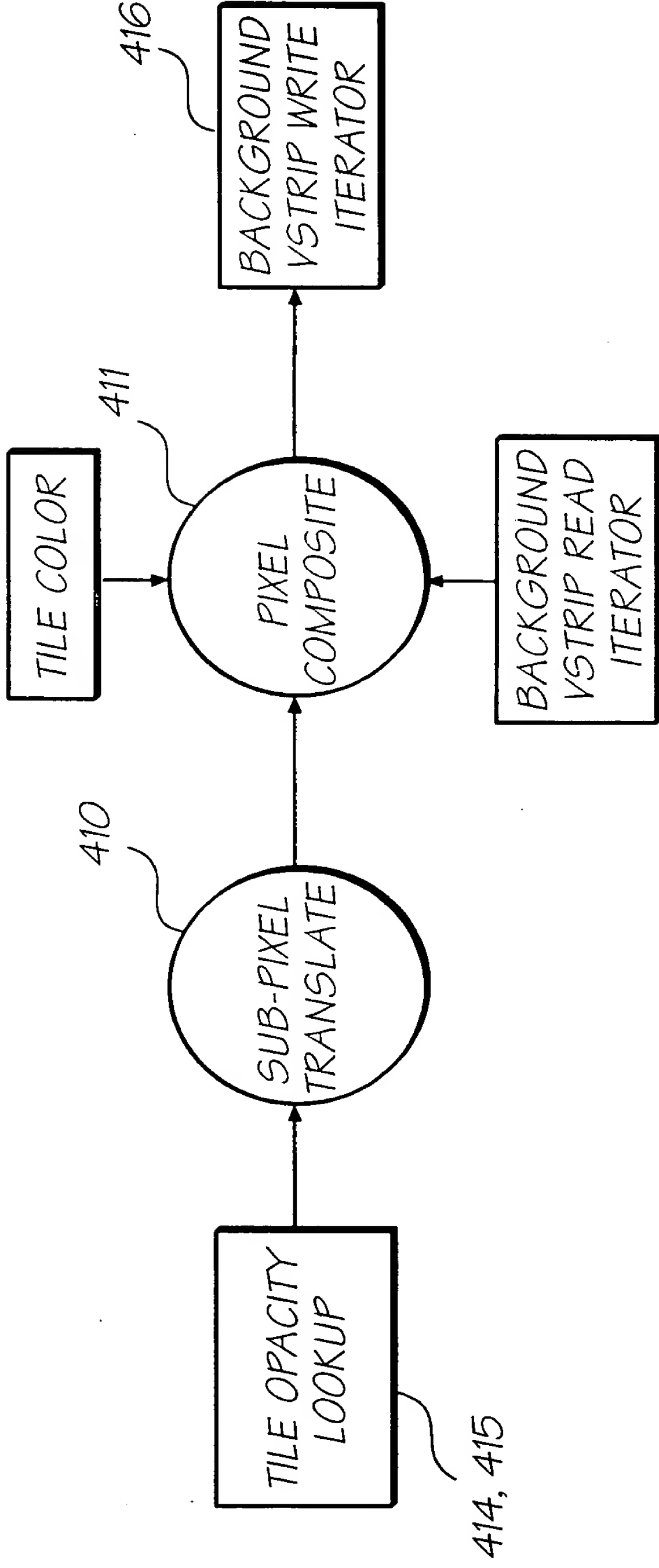


FIG. 103

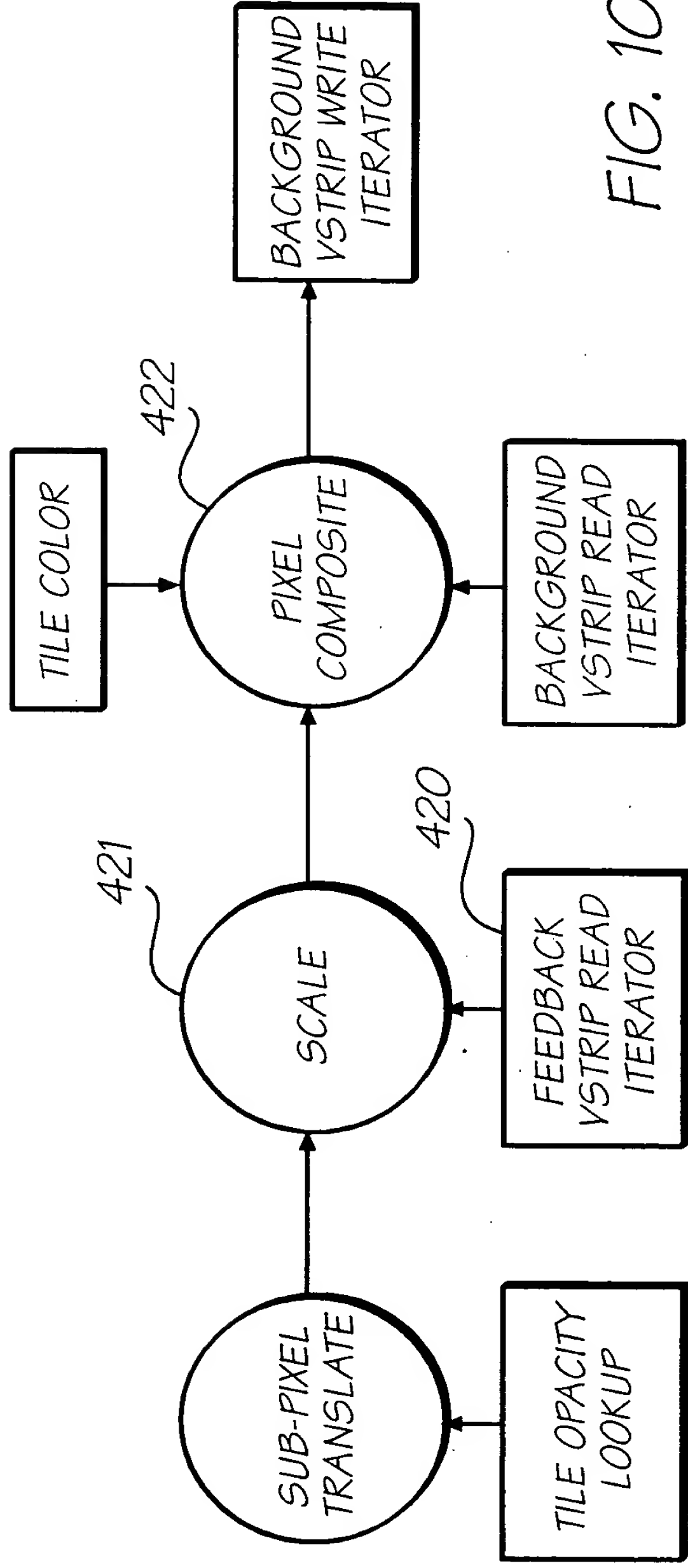


FIG. 104

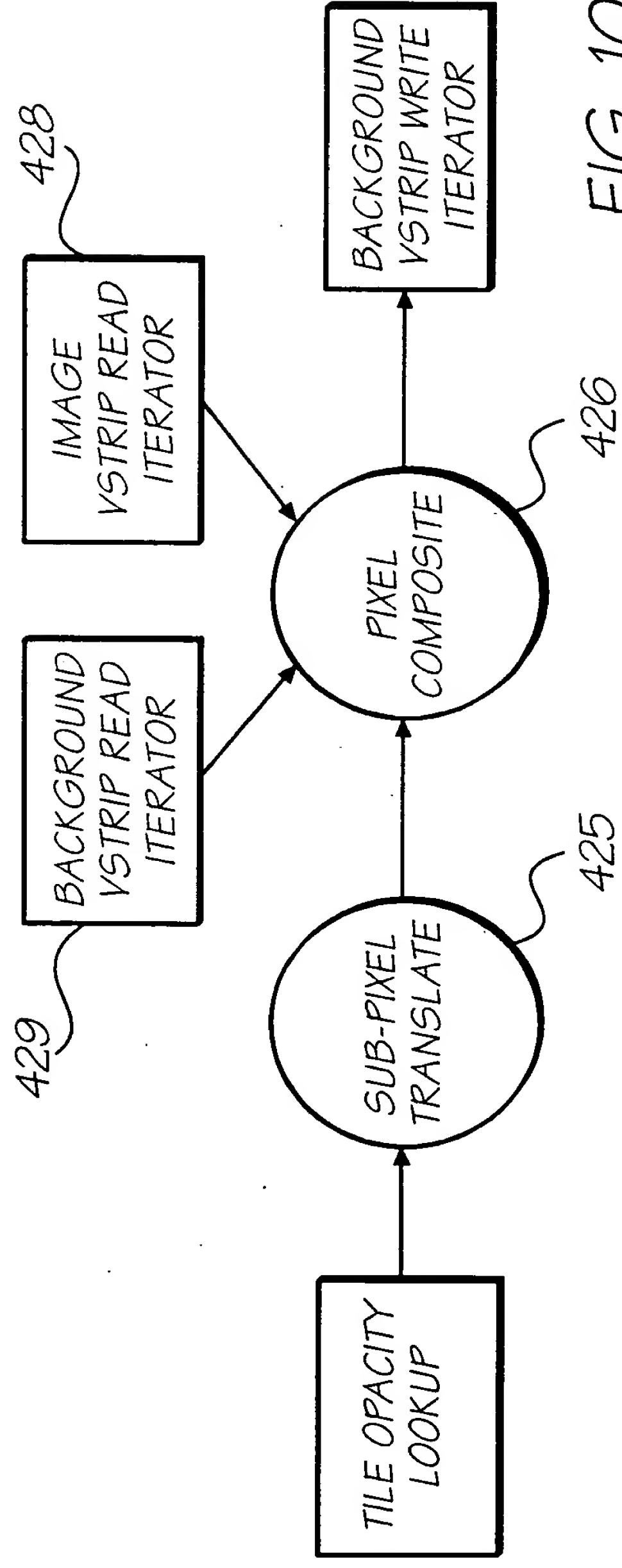


FIG. 105

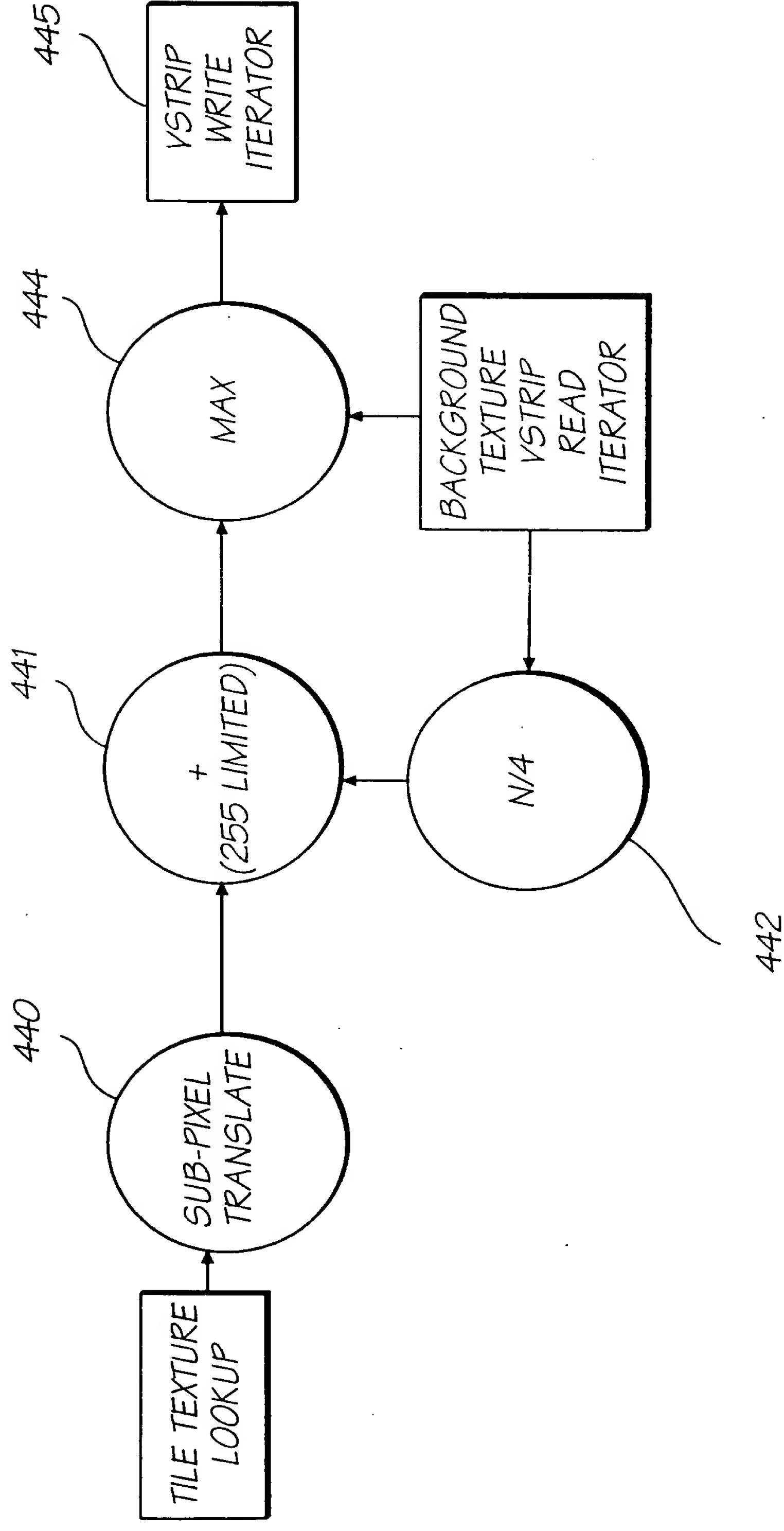


FIG. 108

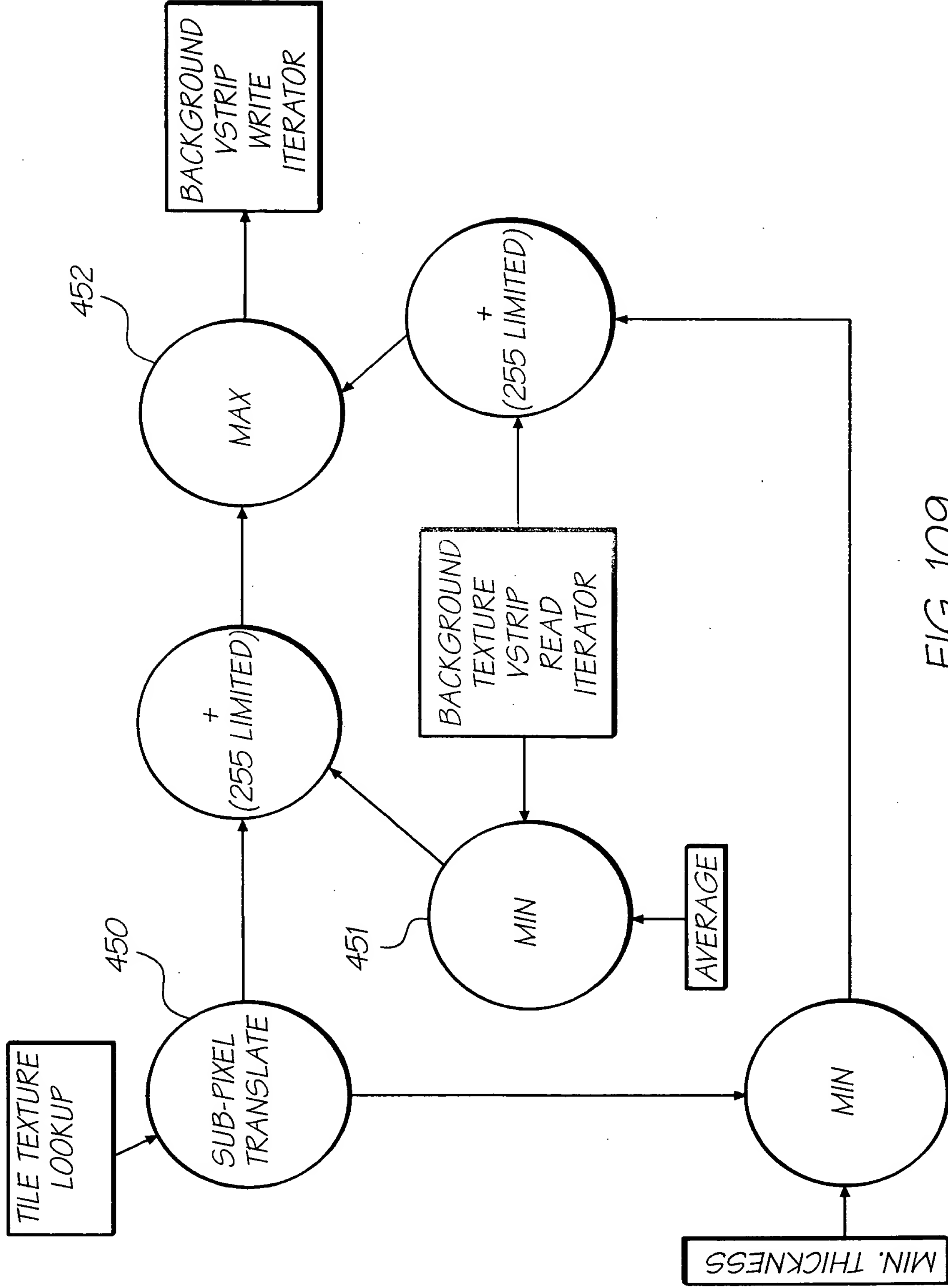


FIG. 109

4604 4603 4602 4601 4600

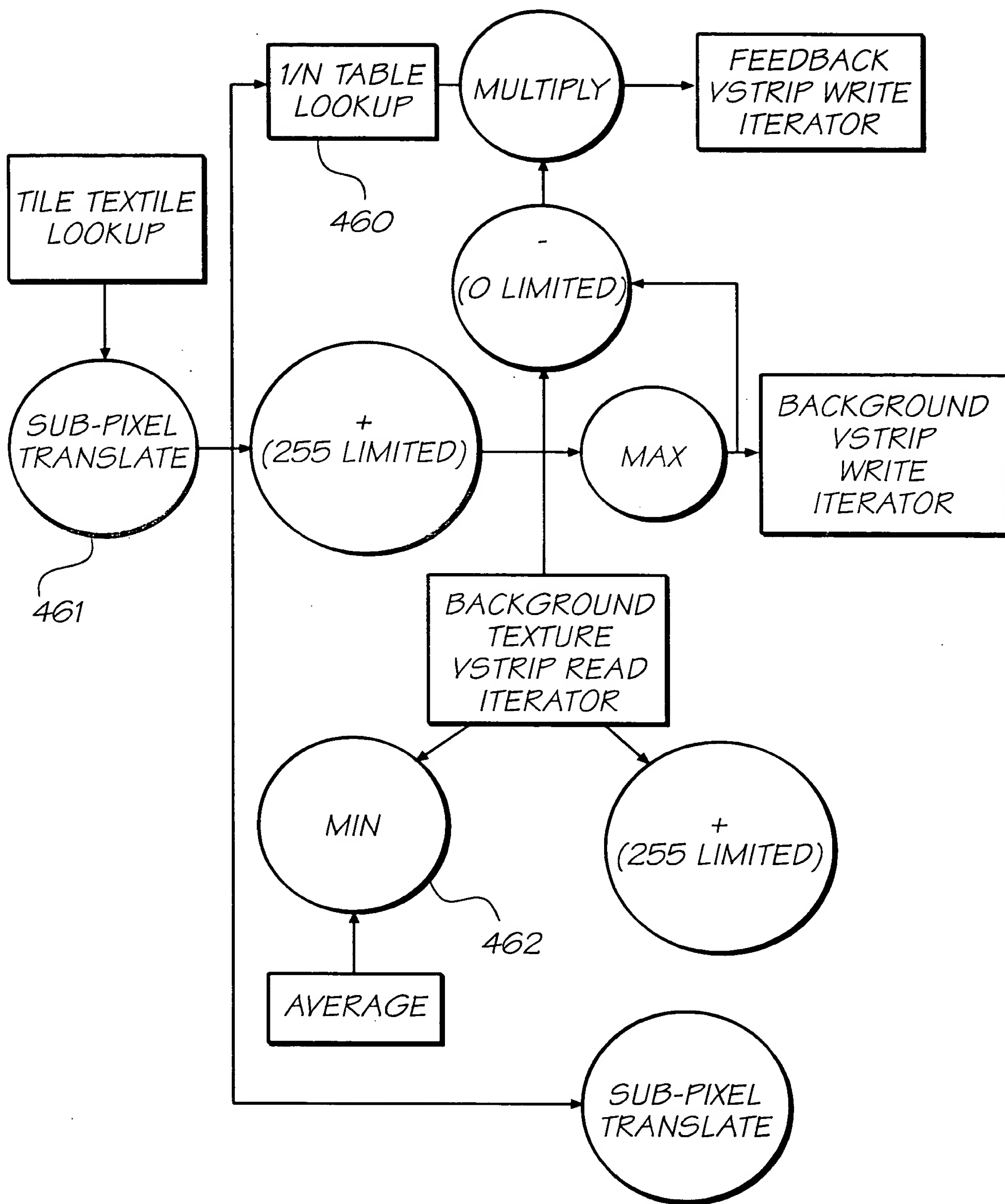
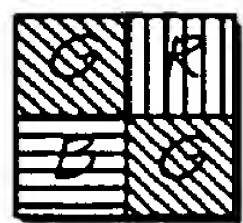


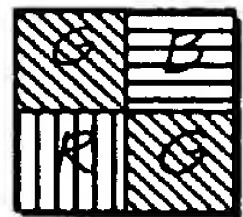
FIG. 110



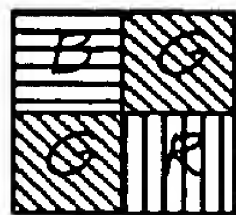
2X2 PIXEL BLOCK,
0 DEGREES



2X2 PIXEL BLOCK,
90 DEGREES

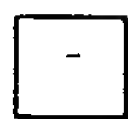
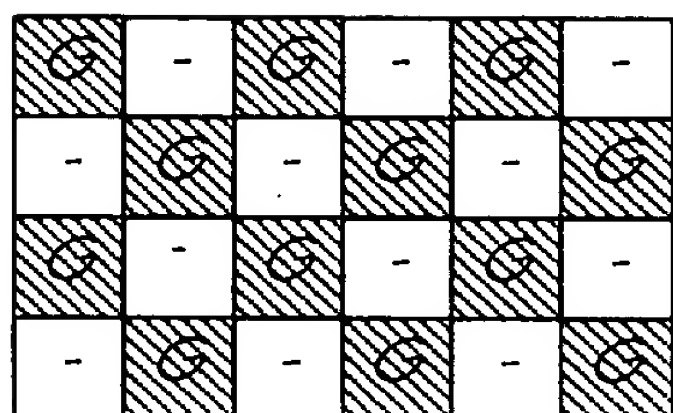


2X2 PIXEL BLOCK,
180 DEGREES



2X2 PIXEL BLOCK,
270 DEGREES

FIG. 111

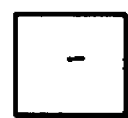
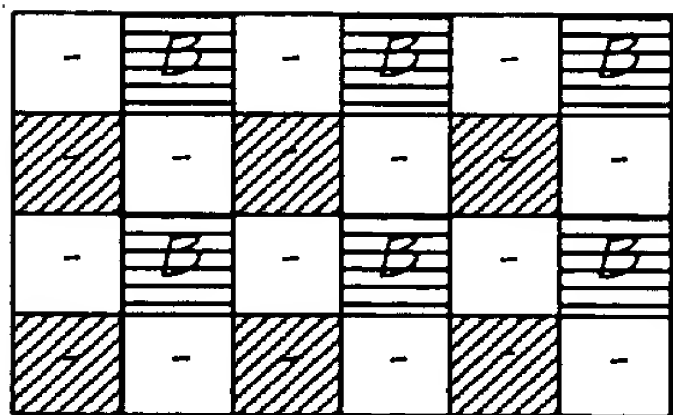


LINEAR INTERPOLATED PIXELS



ACTUAL PIXELS (NOT INTERPOLATED)

FIG. 112



LINEAR INTERPOLATED PIXELS

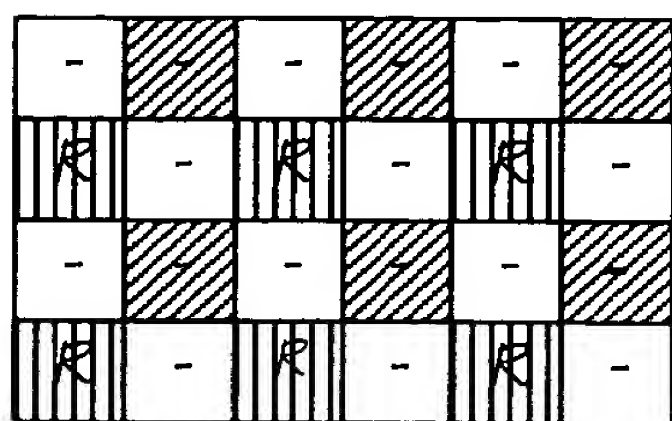


BI-LINEAR INTERPOLATED PIXELS



ACTUAL PIXELS (NOT INTERPOLATED)

FIG. 113



LINEAR INTERPOLATED PIXELS



BI-LINEAR INTERPOLATED PIXELS



ACTUAL PIXELS (NOT INTERPOLATED)

FIG. 114

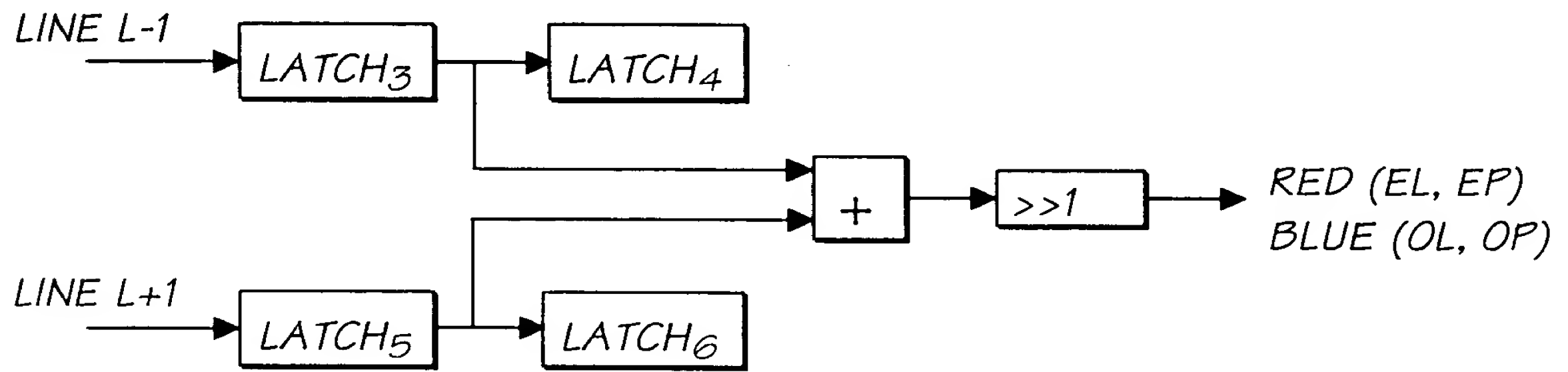
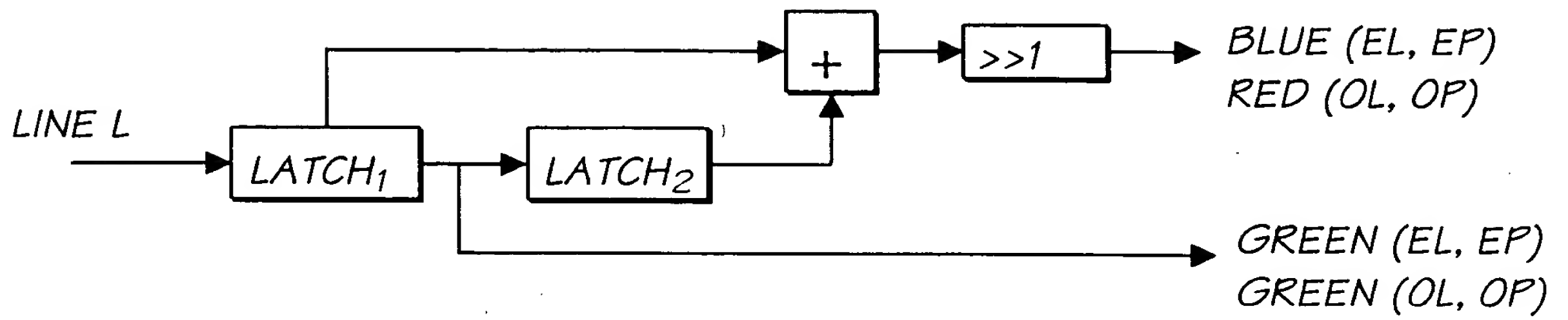


FIG. 115

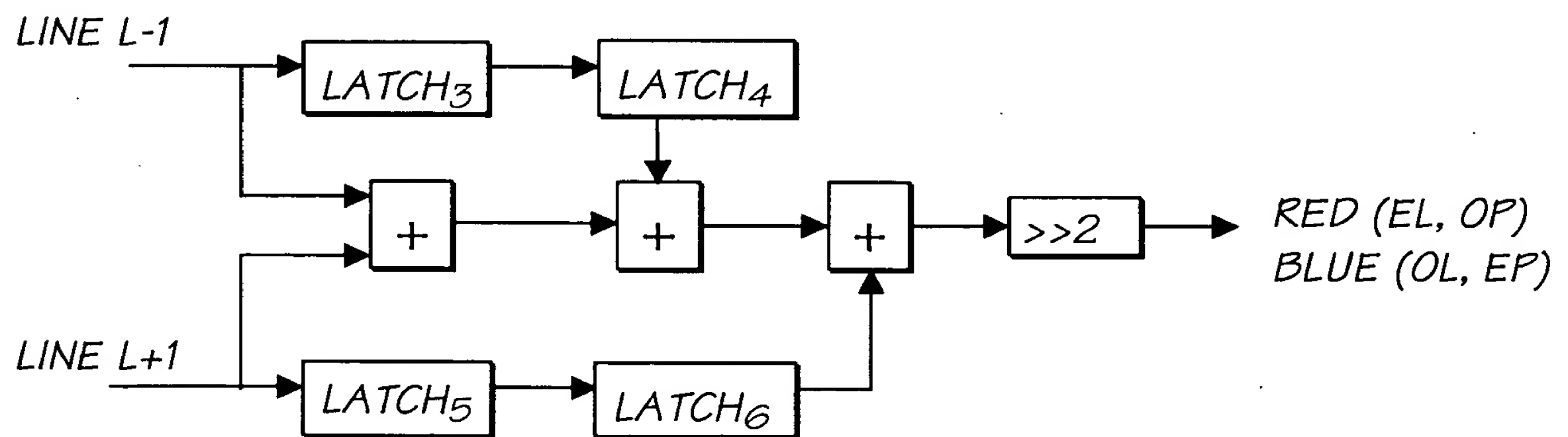
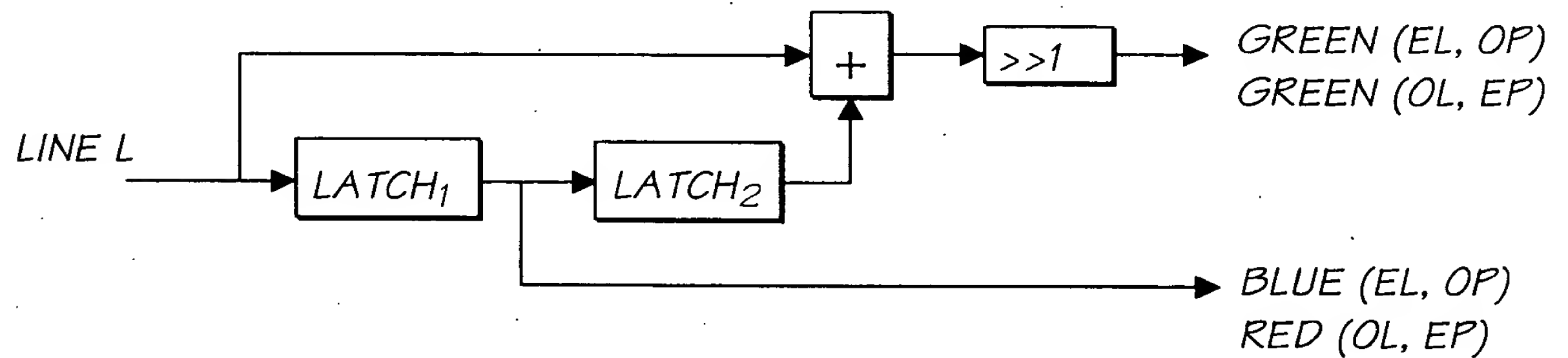


FIG. 116

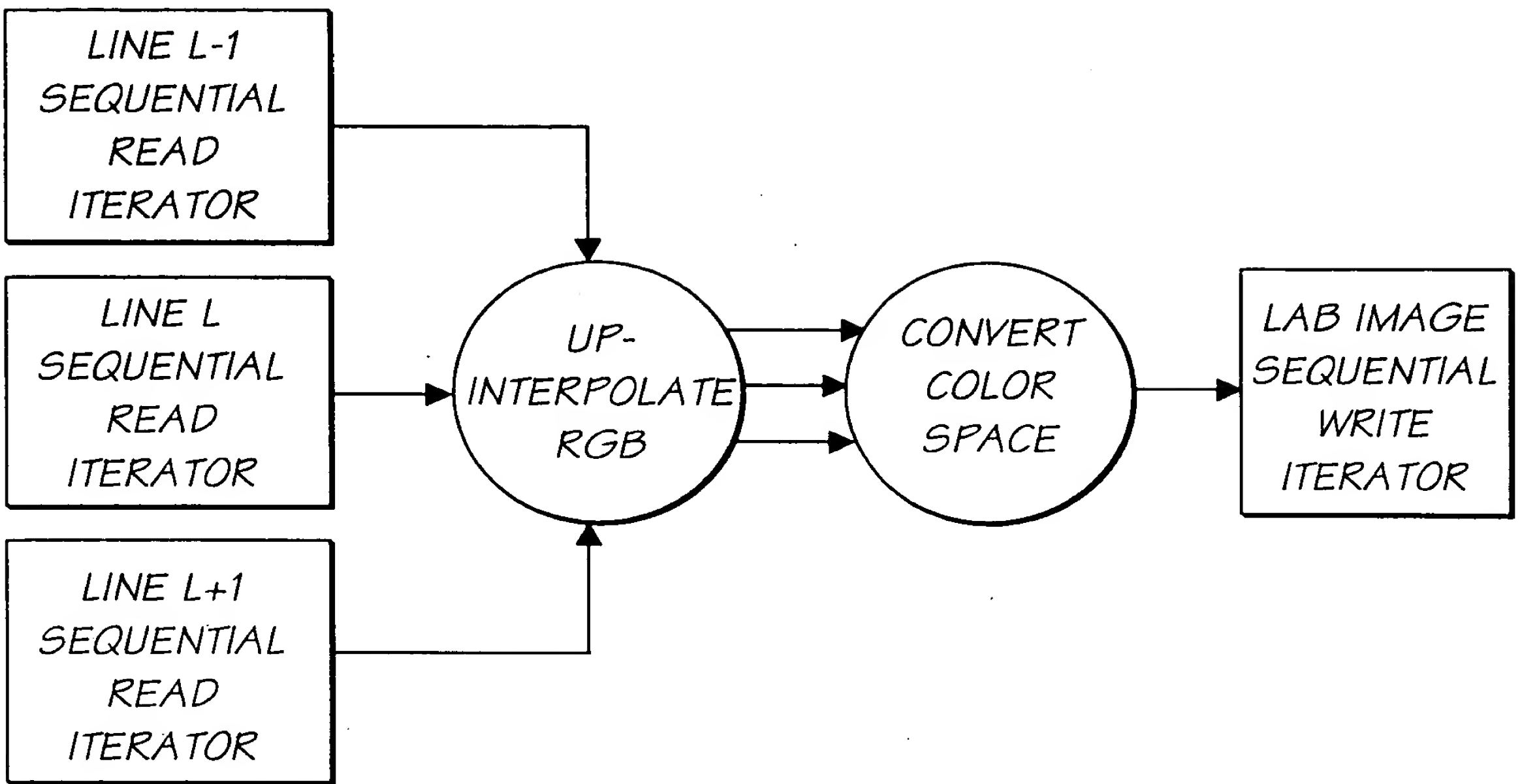


FIG. 117

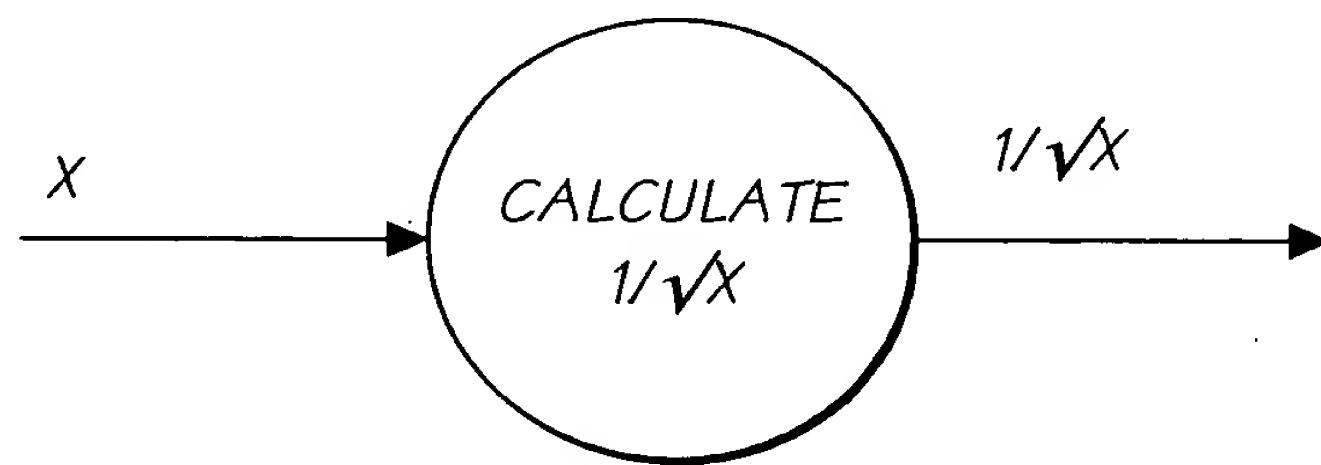


FIG. 118

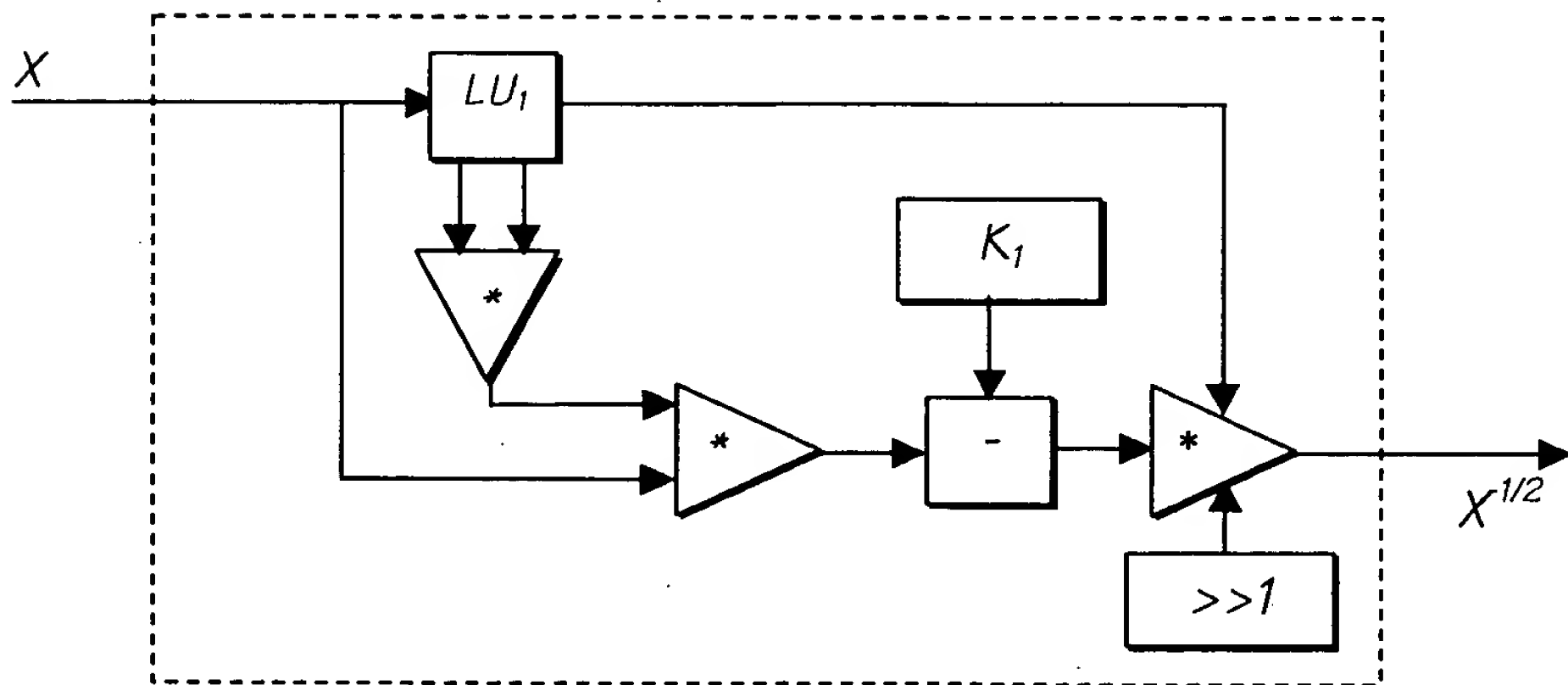


FIG. 119

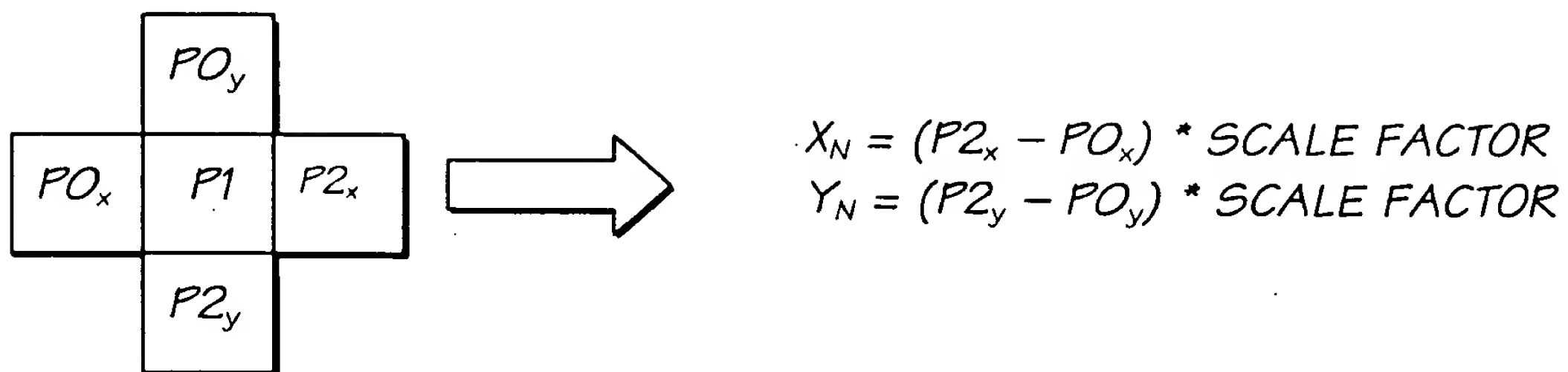


FIG. 120

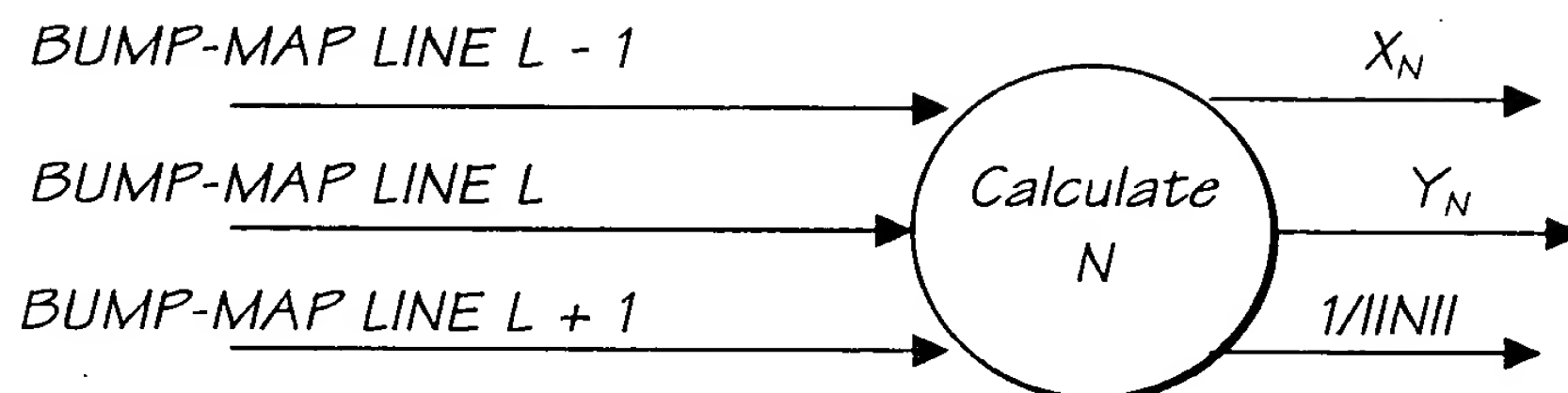


FIG. 121

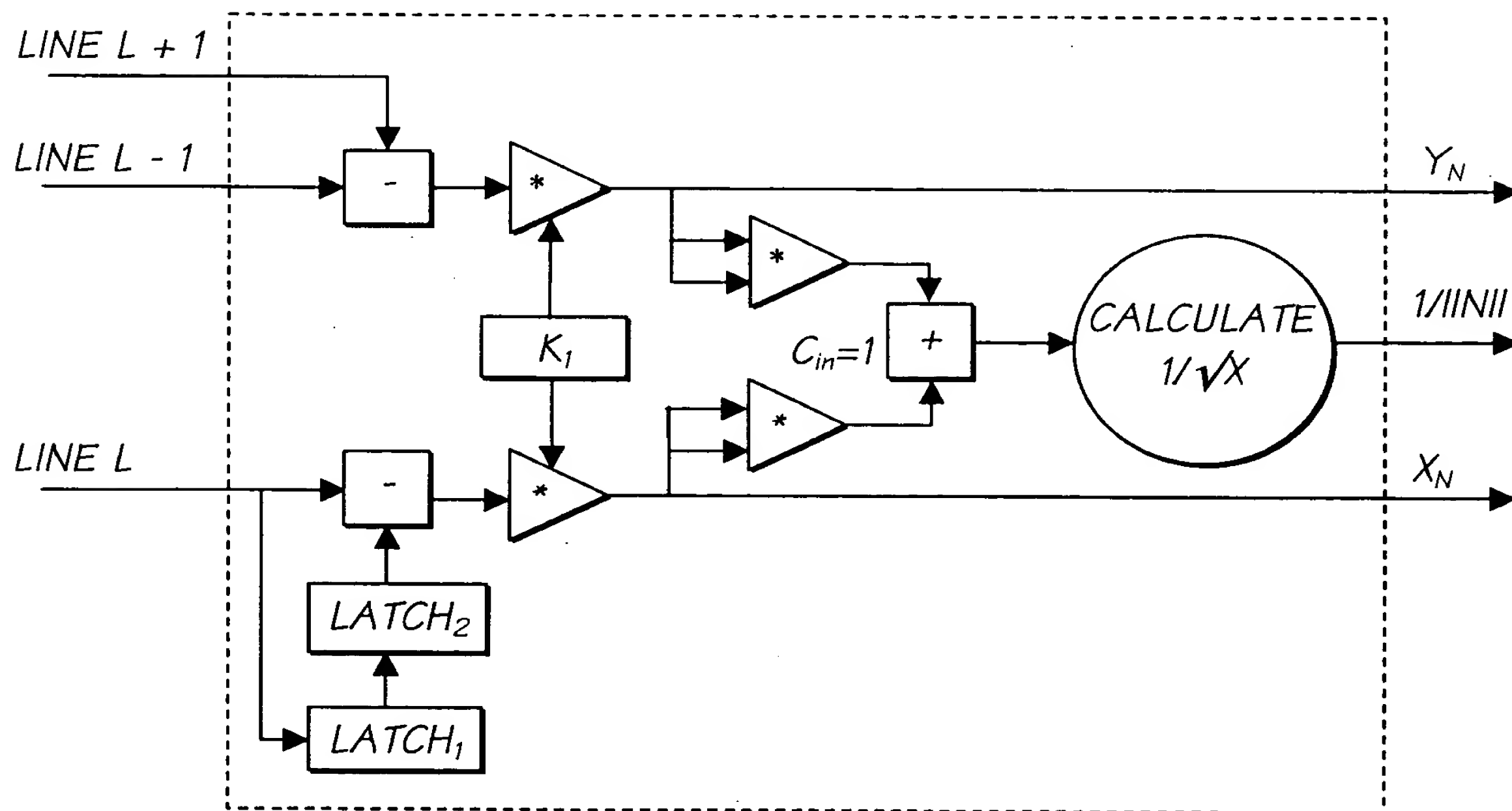


FIG. 122

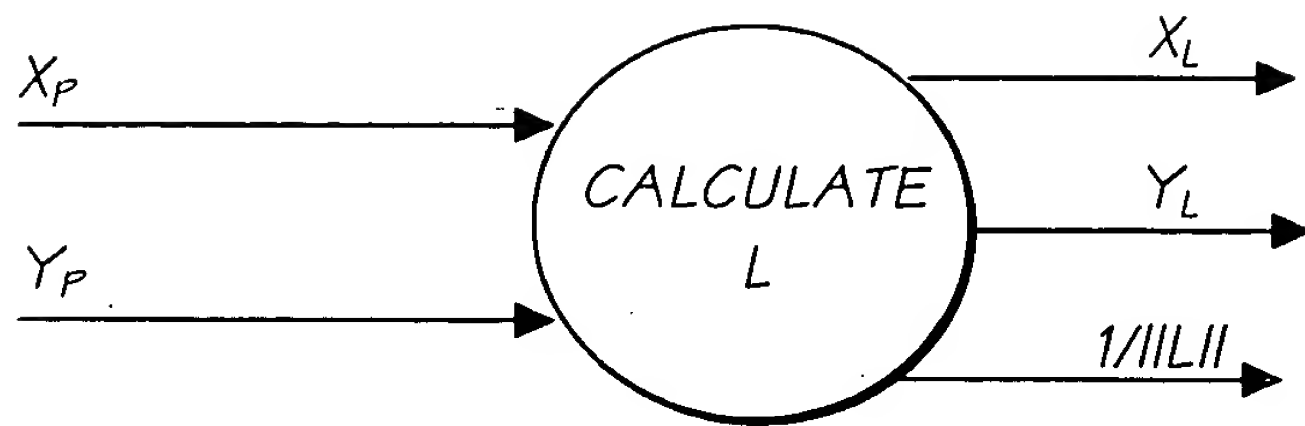


FIG. 123

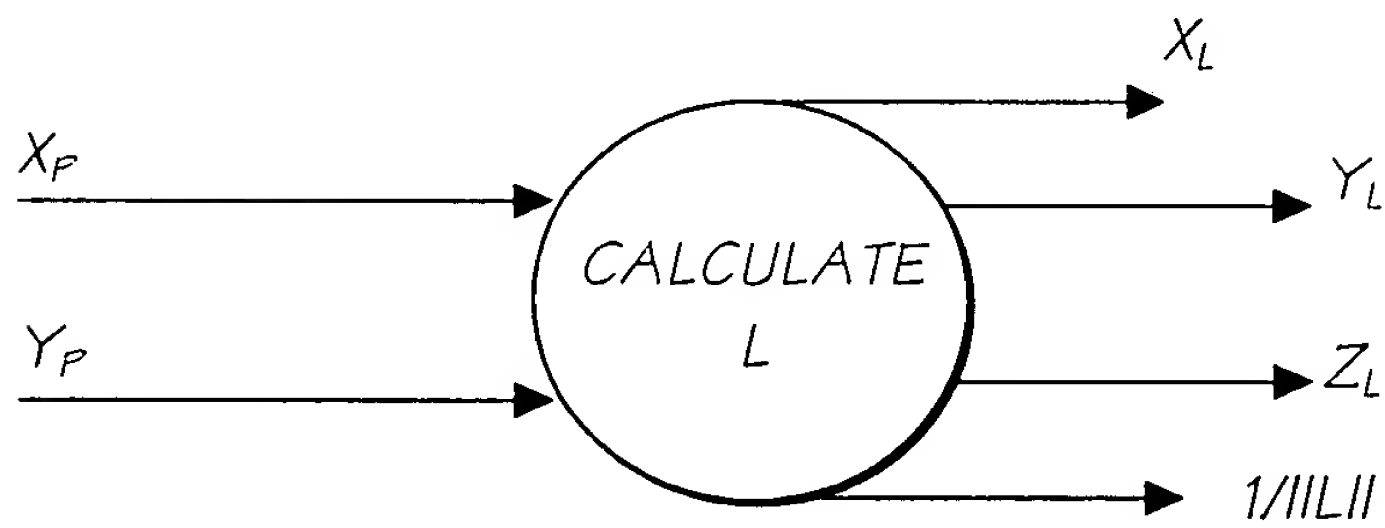


FIG. 124

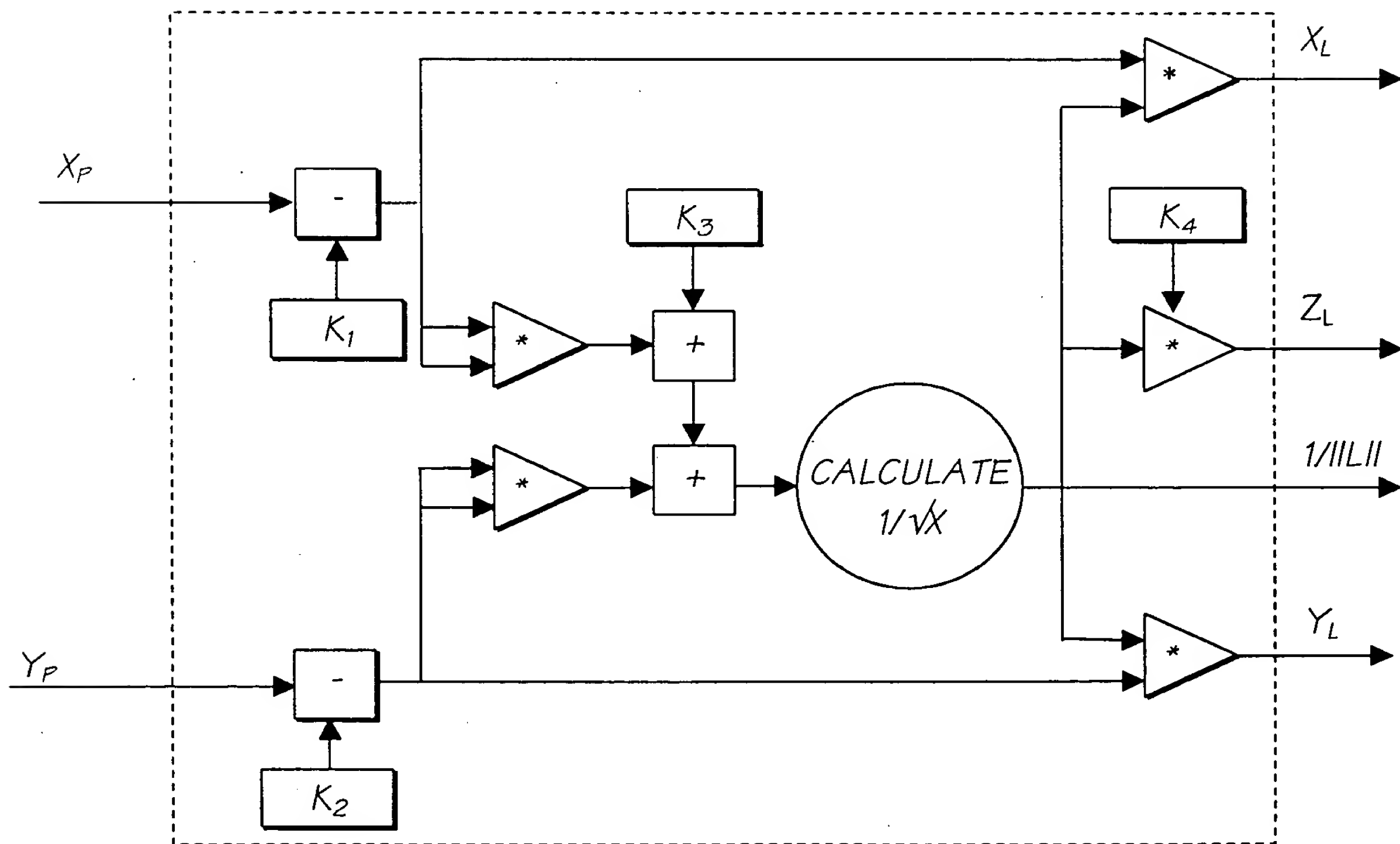


FIG. 125

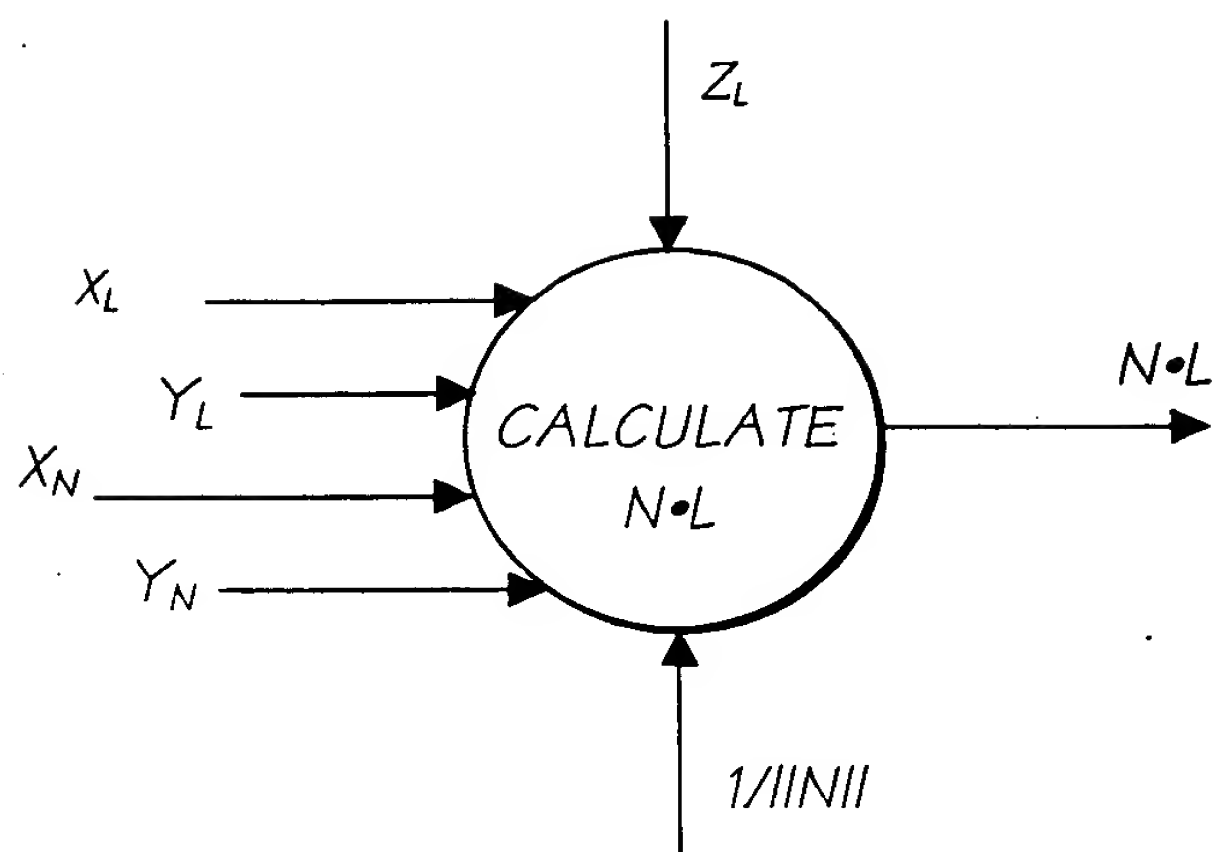


FIG. 126

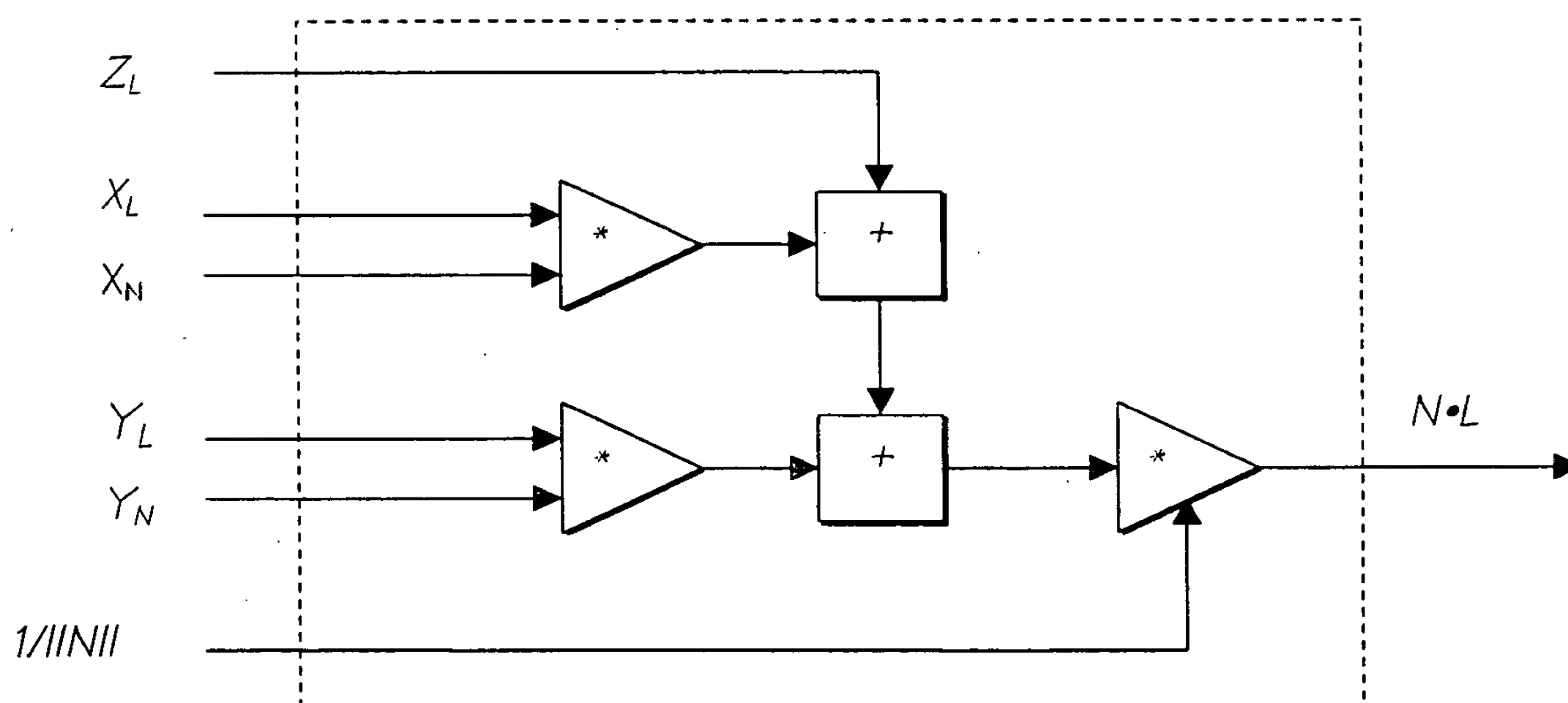


FIG. 127

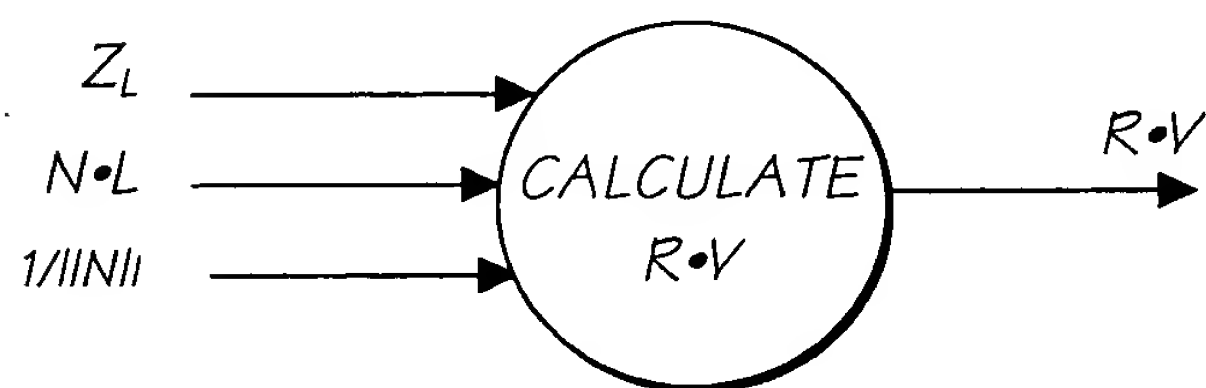


FIG. 128

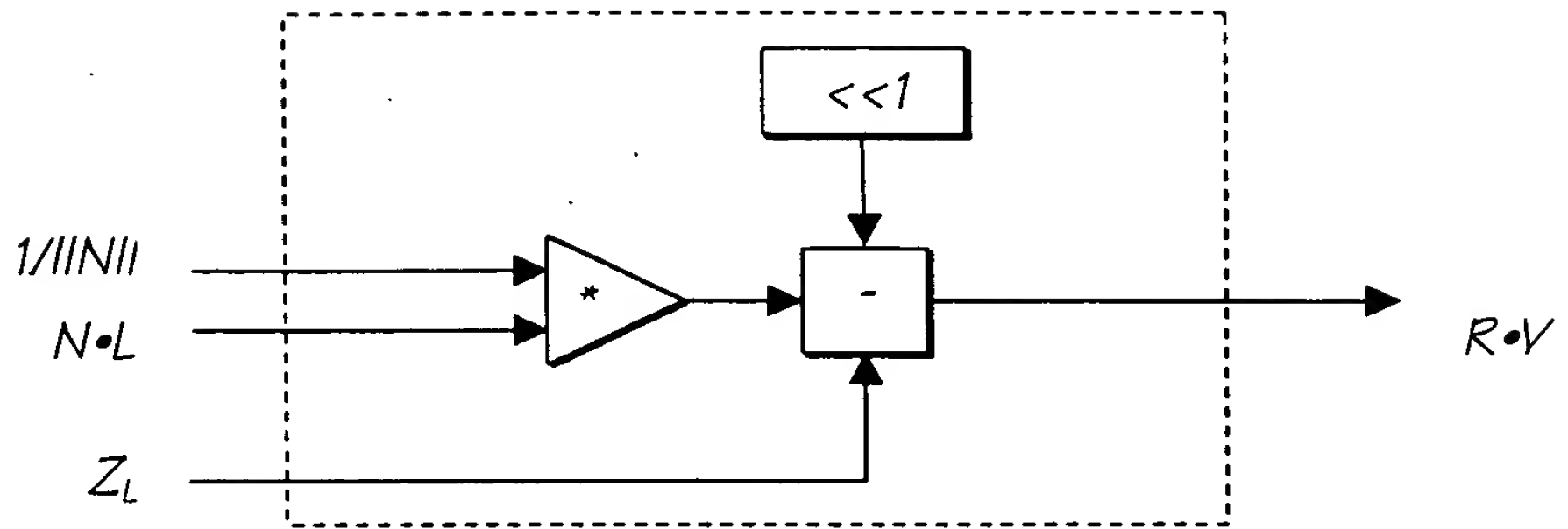


FIG. 129

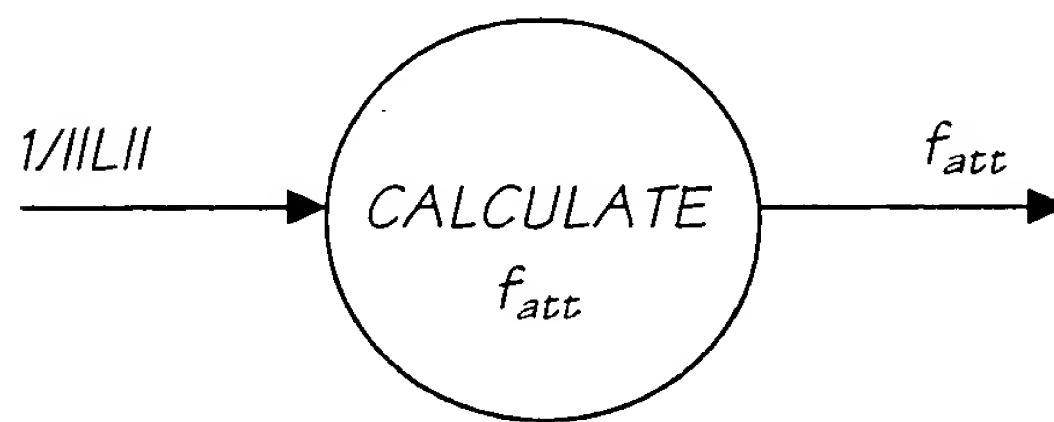


FIG. 130

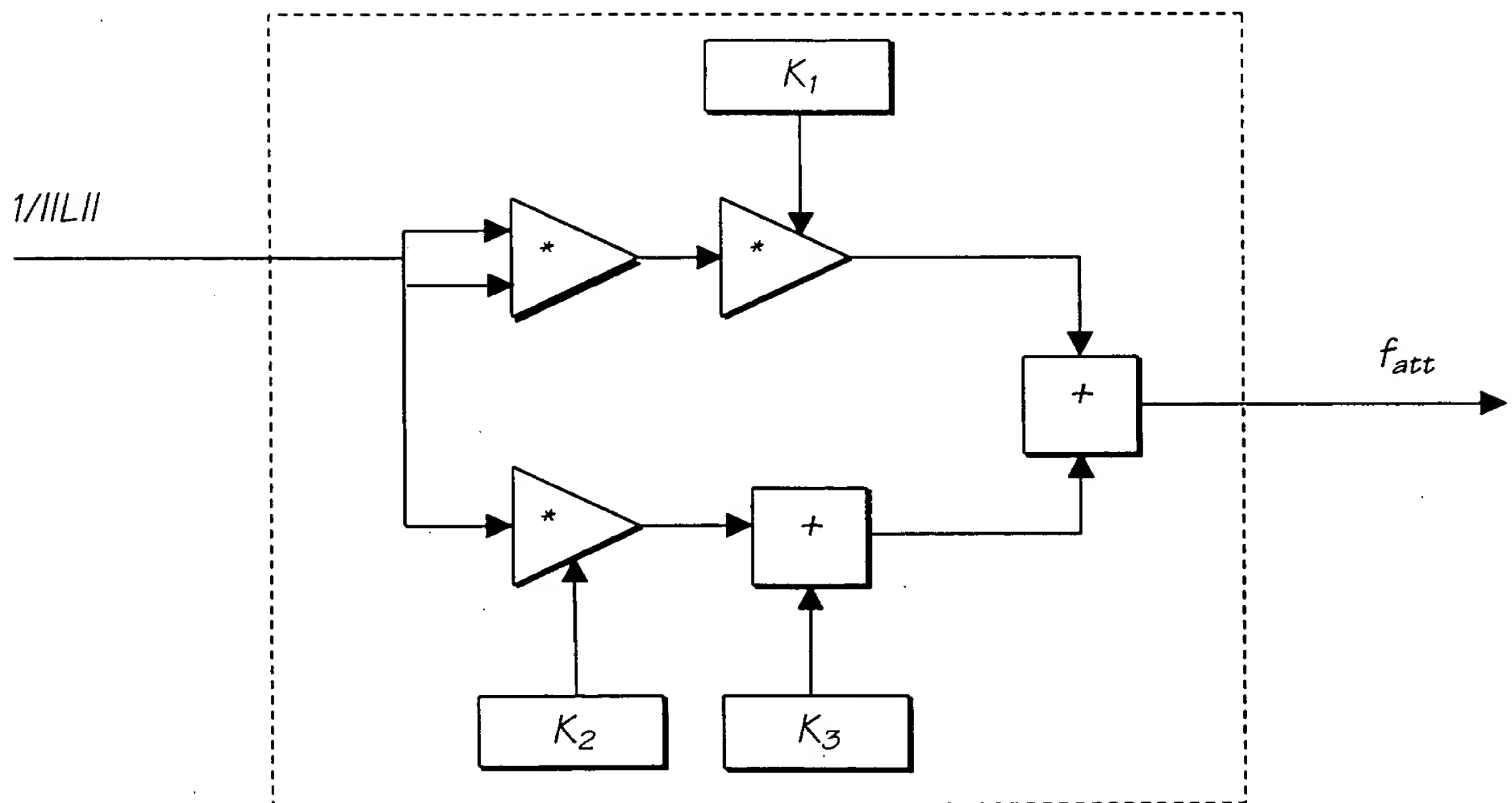


FIG. 131

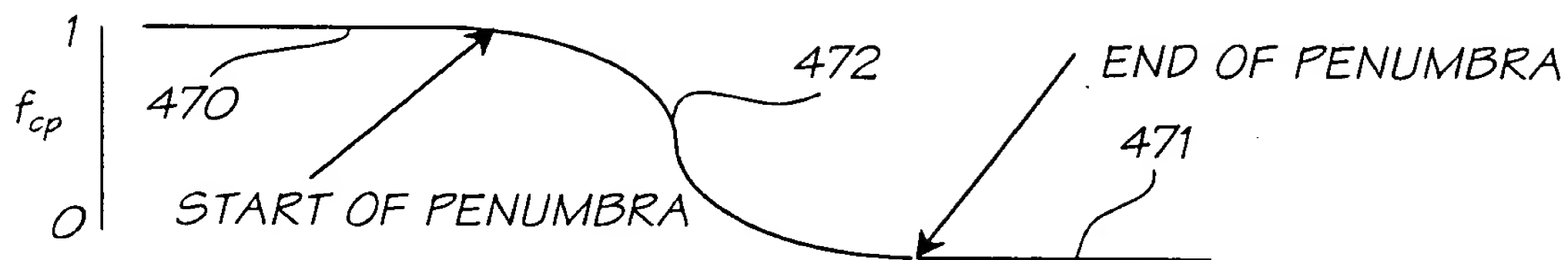


FIG. 132

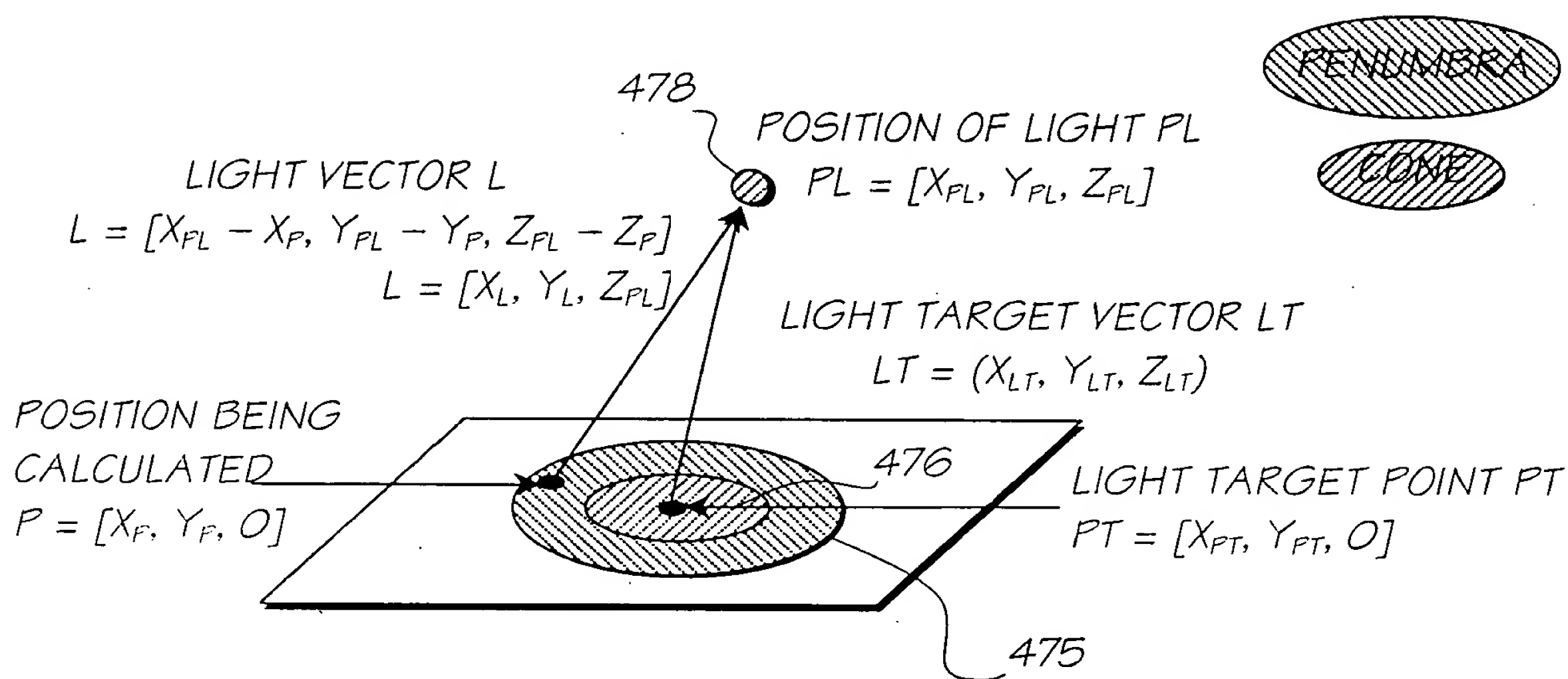


FIG. 133

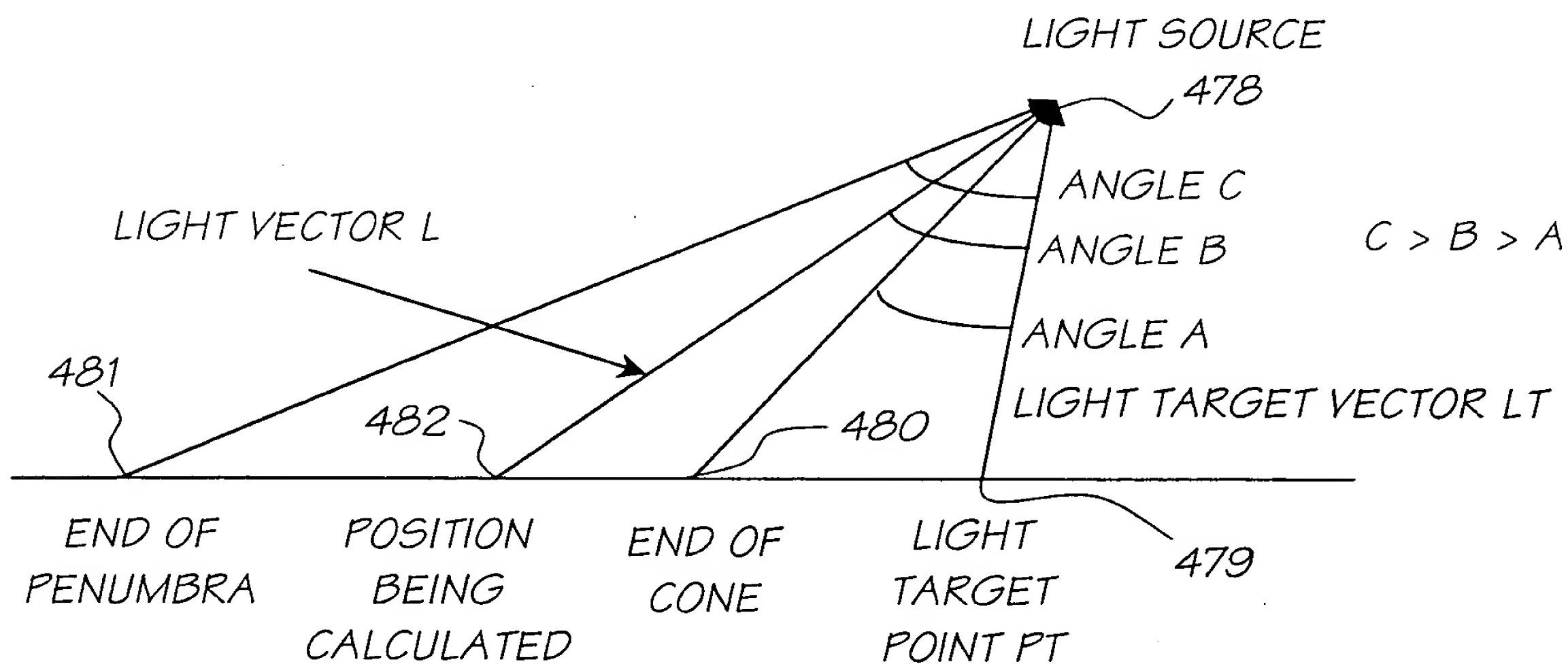


FIG. 134

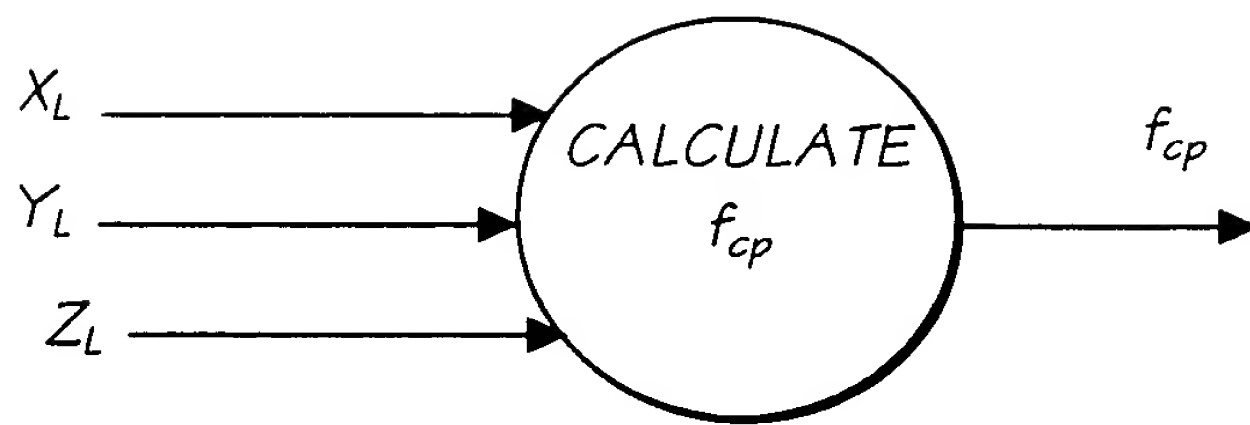


FIG. 135

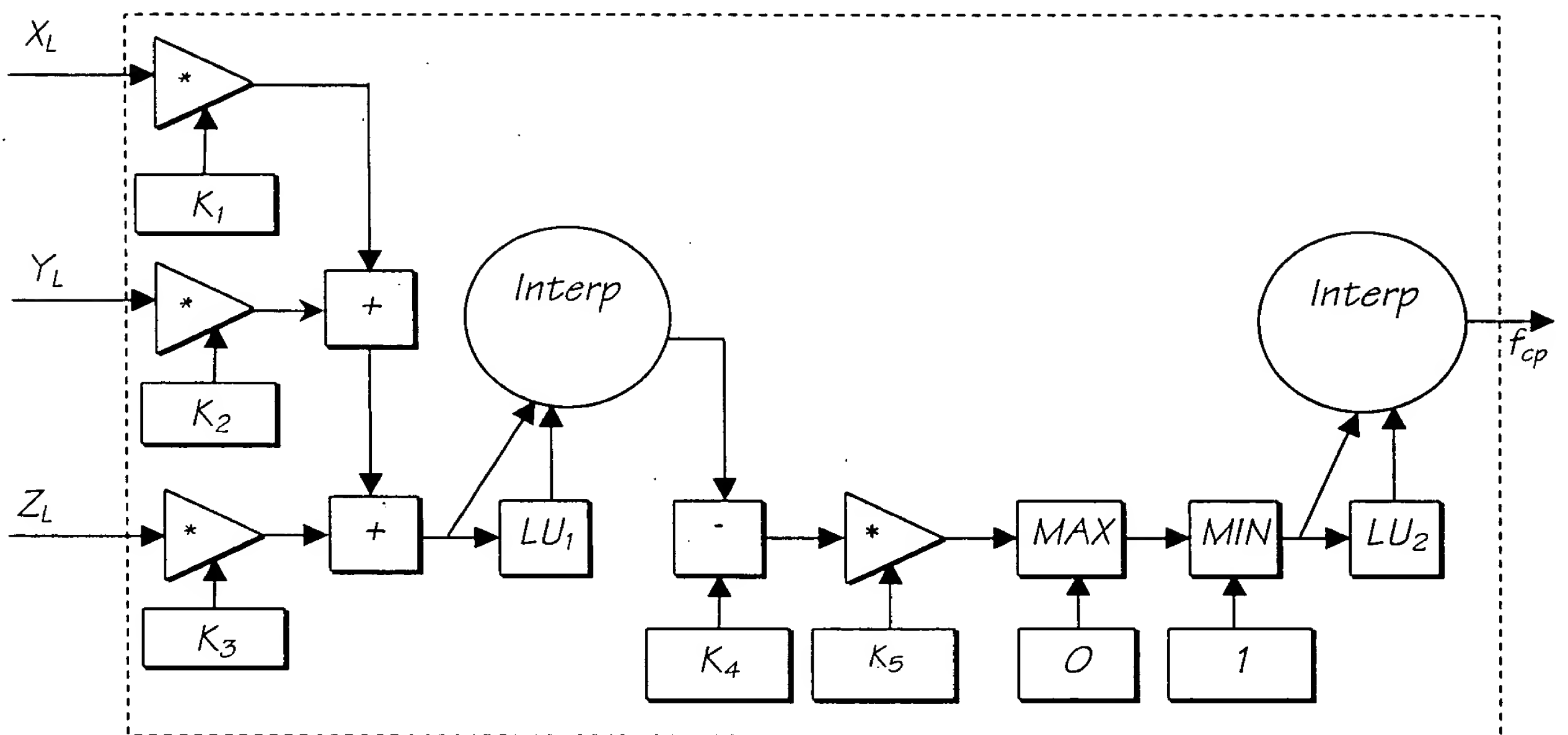


FIG. 136

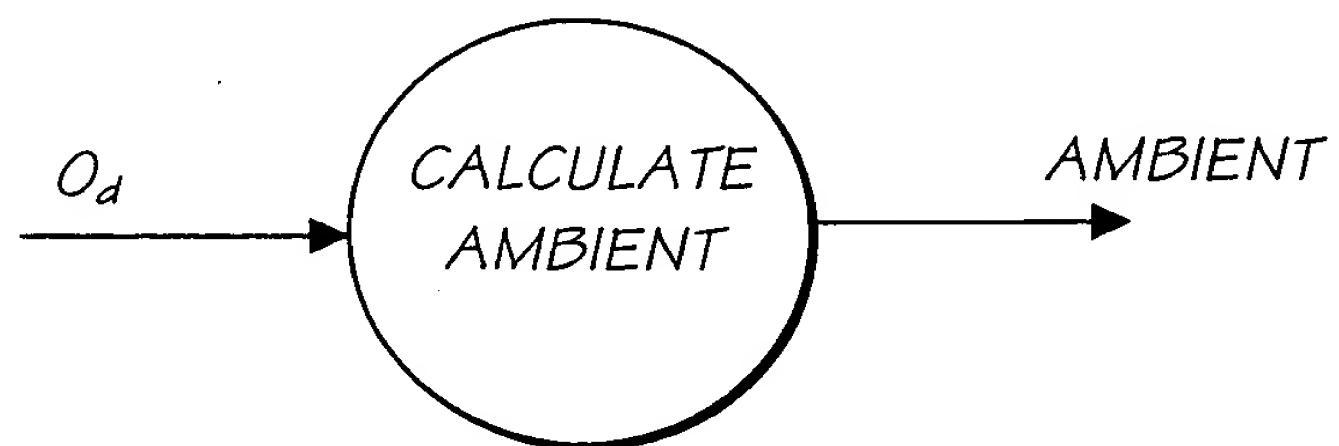


FIG. 137

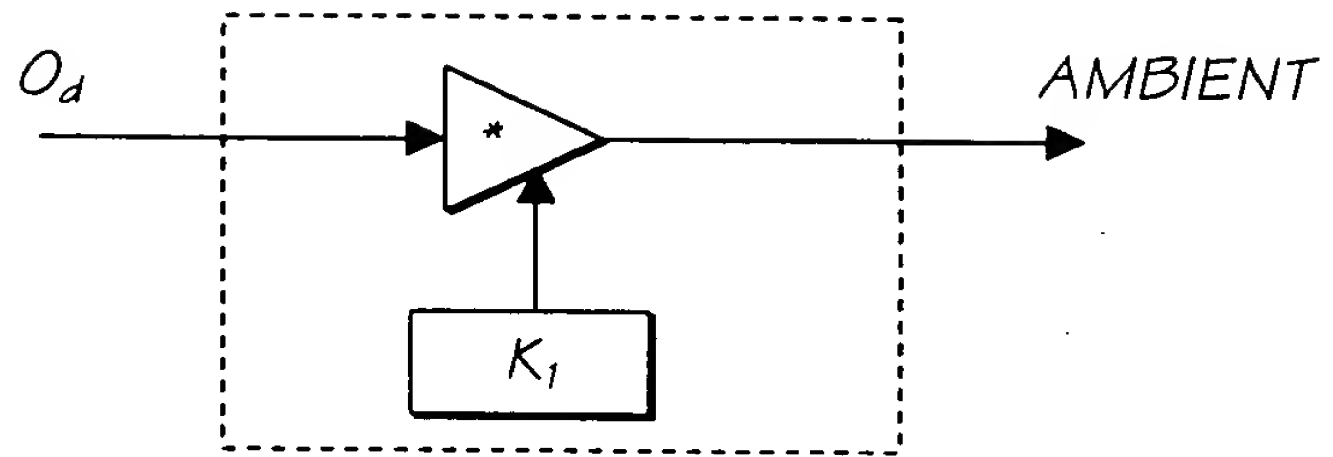


FIG. 138

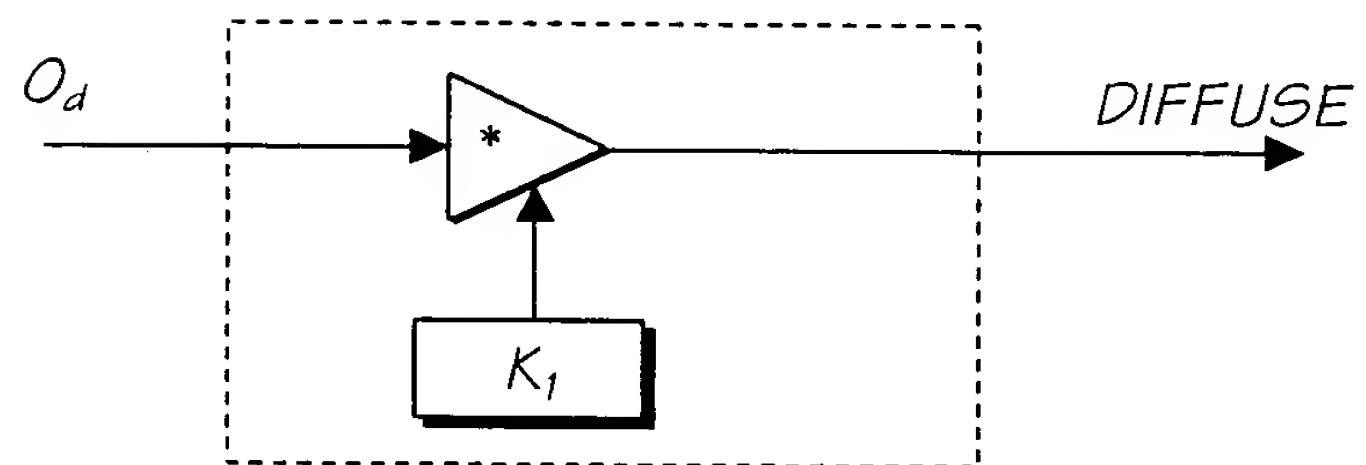


FIG. 139

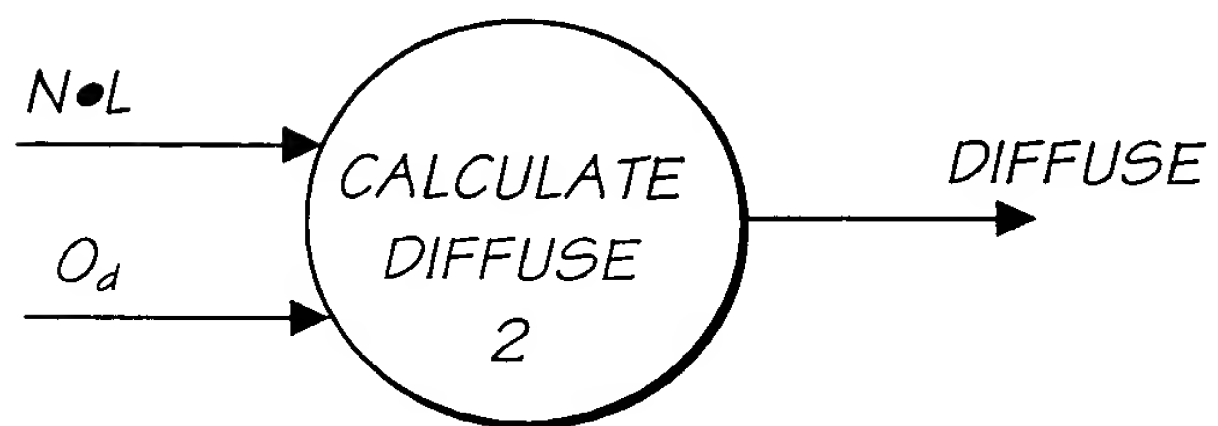


FIG. 140

2025 RELEASE UNDER E.O. 14176

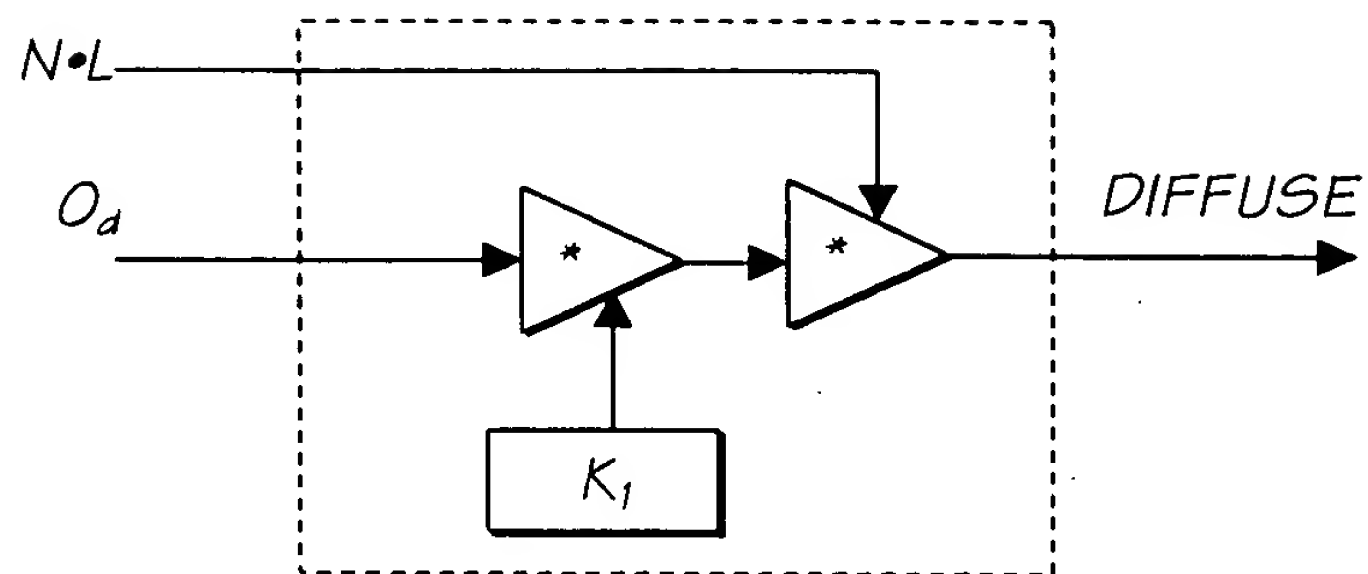


FIG. 141

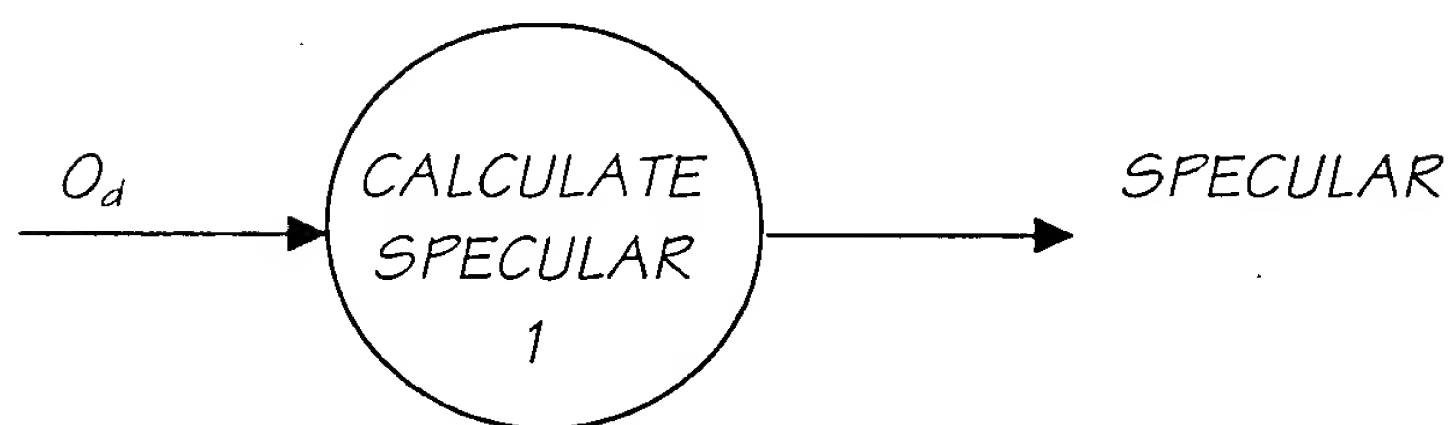


FIG. 142

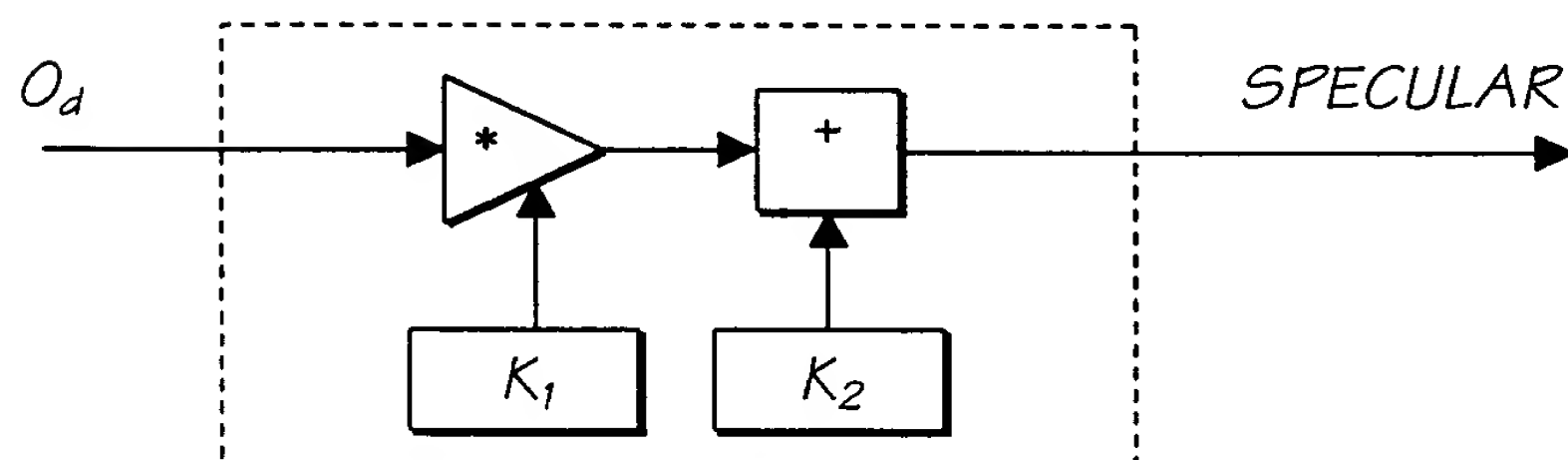


FIG. 143

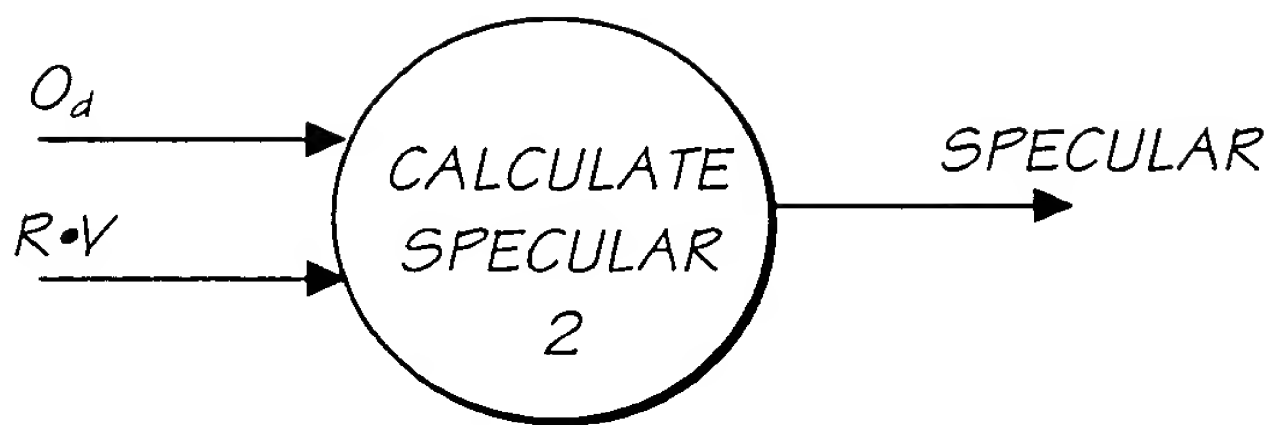


FIG. 144

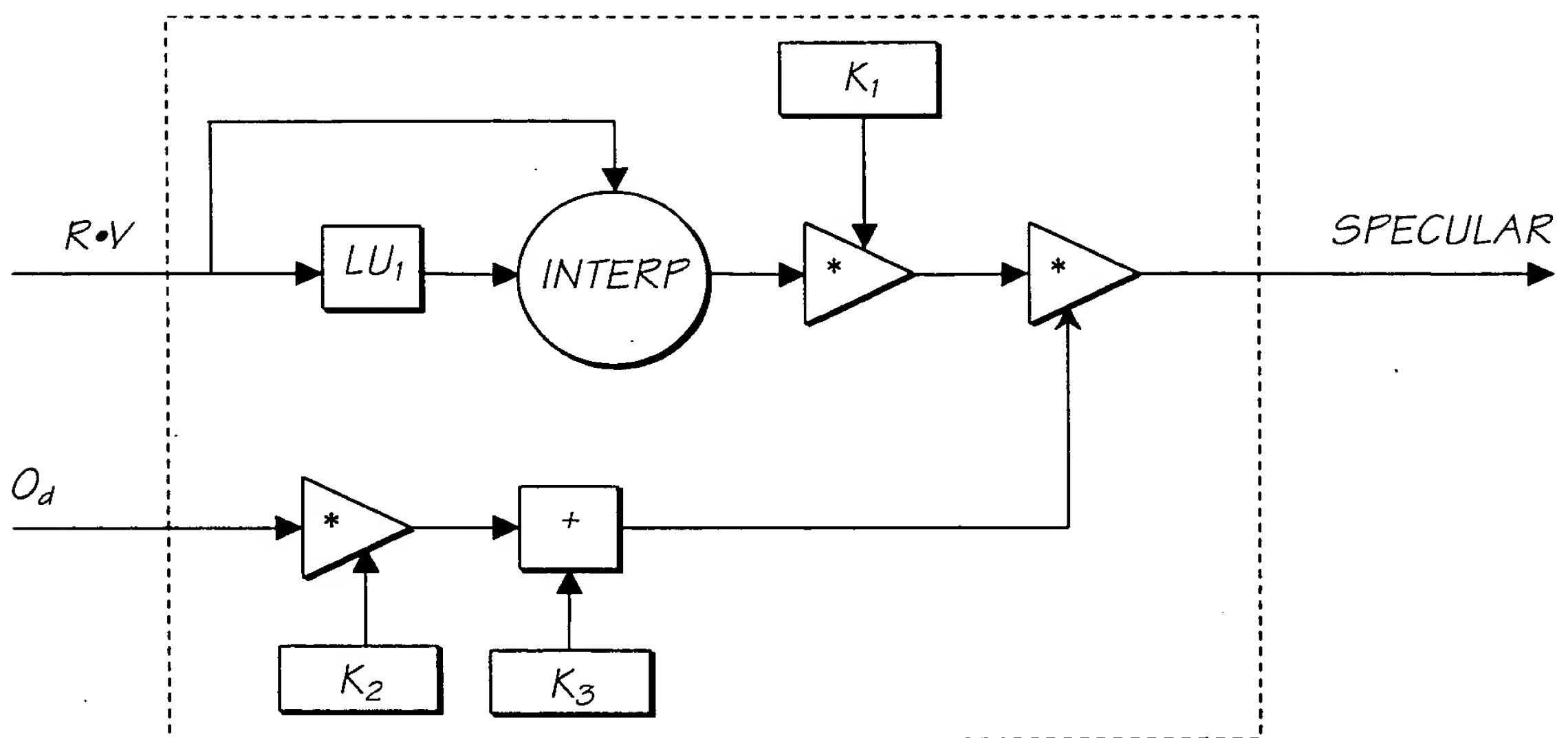


FIG. 145

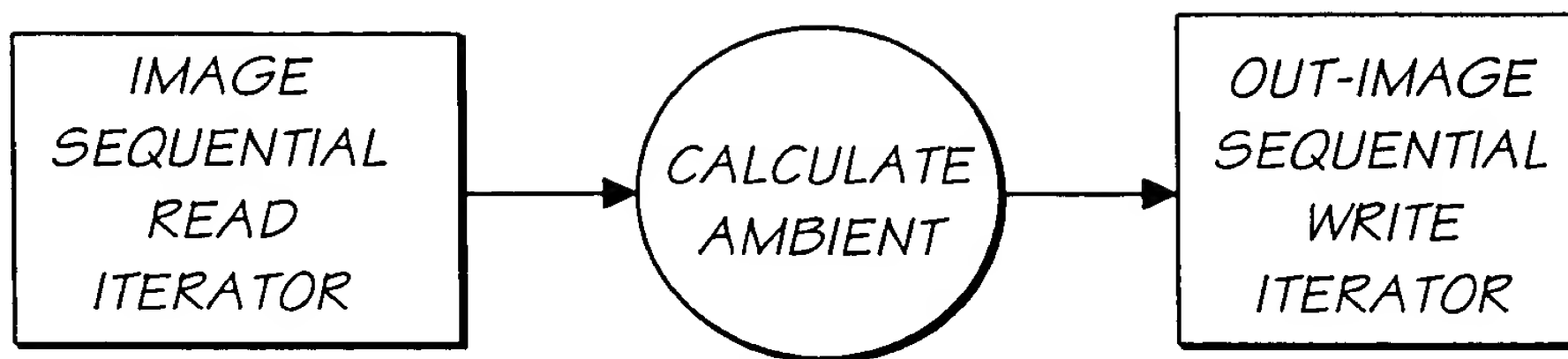
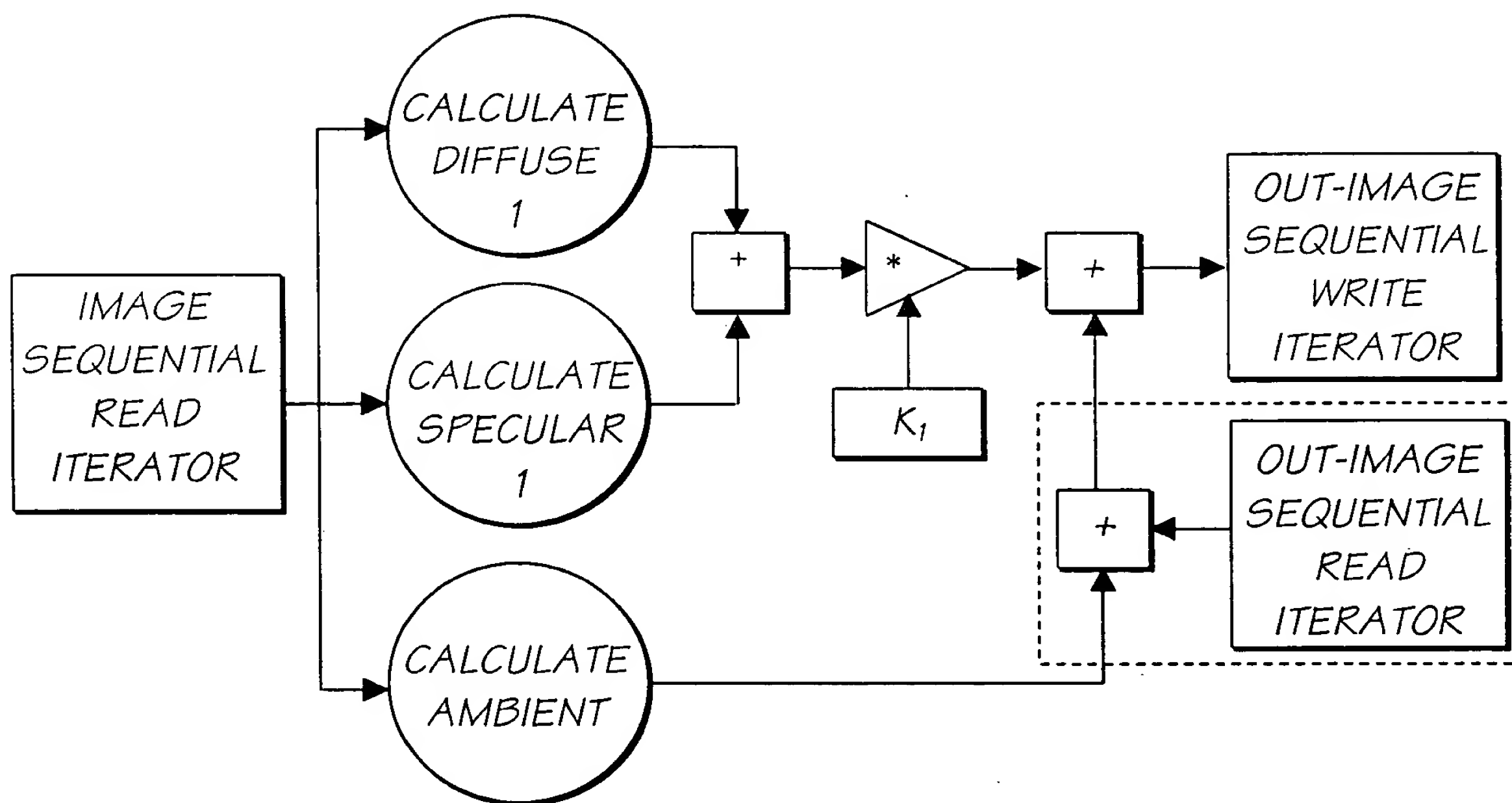


FIG. 146



 2ND AND
SUBSEQUENT LIGHTS

FIG. 147

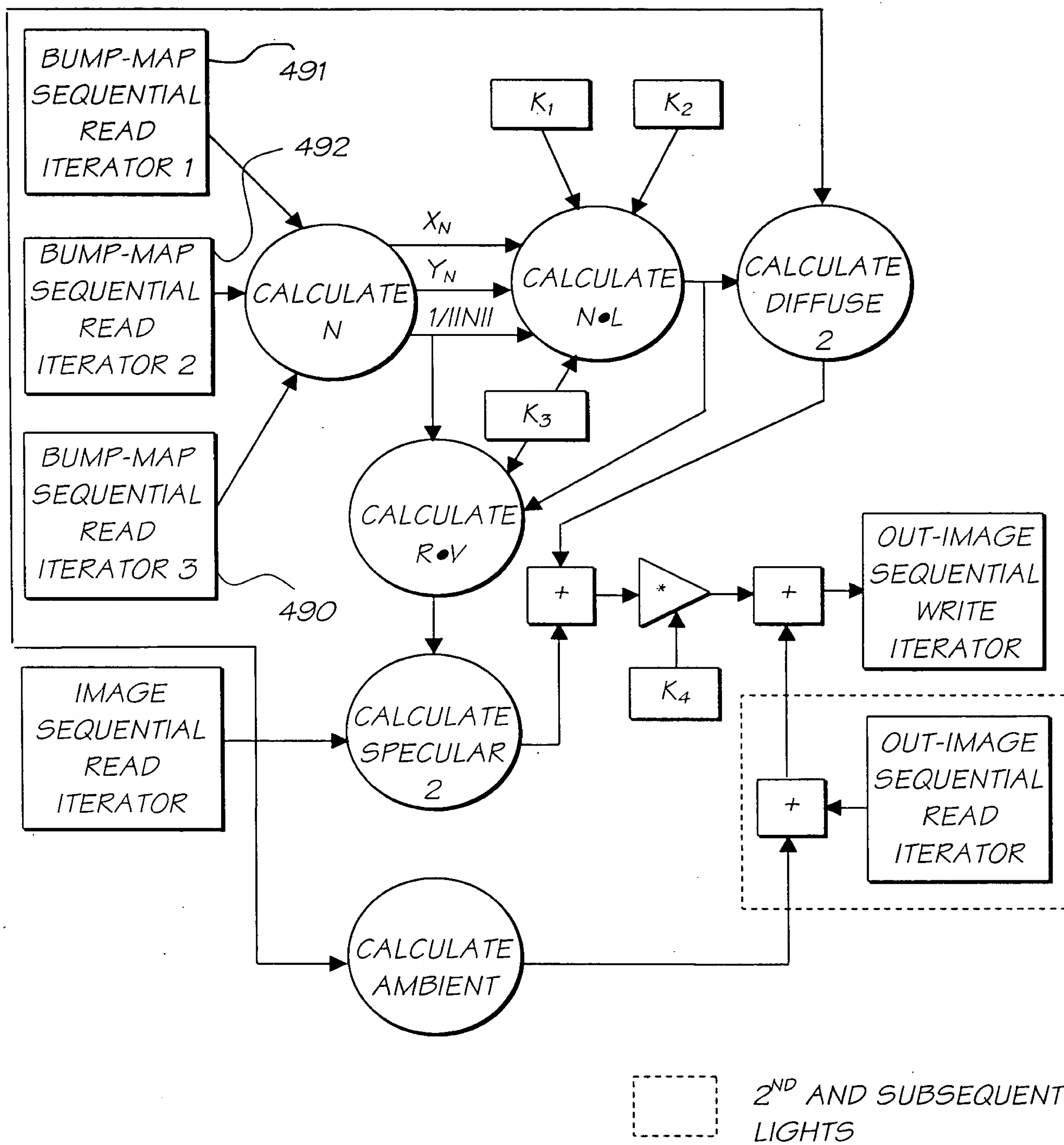


FIG. 148

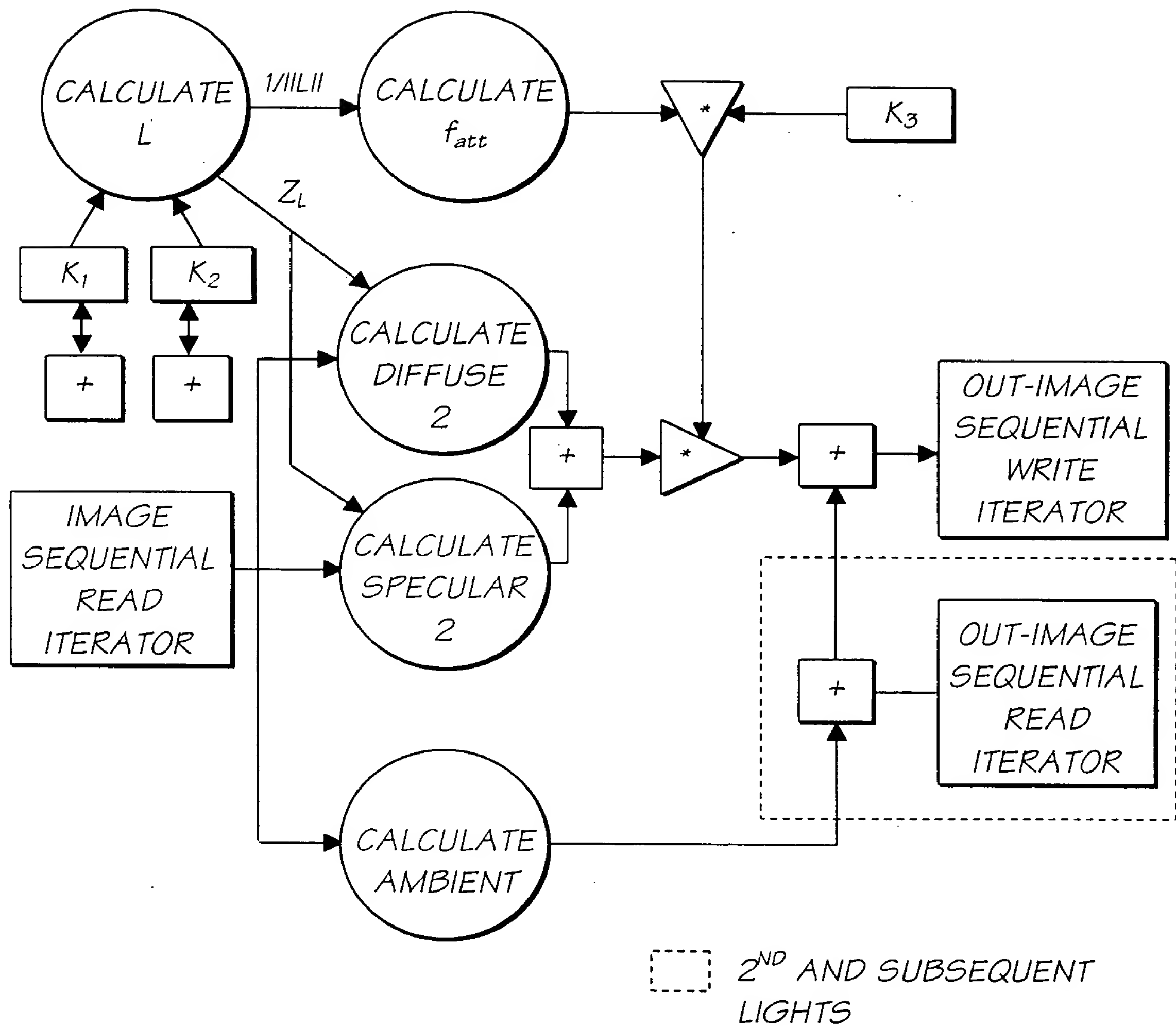


FIG. 149

The flowchart illustrates a ray-tracing algorithm for global illumination, including recursive calls for subsequent lights. The process begins with three parallel inputs: **BUMP-MAP SEQUENTIAL READ ITERATOR 1**, **BUMP-MAP SEQUENTIAL READ ITERATOR 2**, and **BUMP-MAP SEQUENTIAL READ ITERATOR 3**. These feed into a circular node labeled **CALCULATE N**. This node also receives a feedback signal $1/|I|N|I|$ and outputs X_N and Y_N to a **CALCULATE N•L** node. The **CALCULATE N•L** node also receives X_L , Y_L , and Z_L from a **CALCULATE L** node. The **CALCULATE L** node receives inputs from two adders, each preceded by a gain block K_1 and K_2 respectively. The **CALCULATE L** node outputs to a **CALCULATE f_{att}** node. The **CALCULATE f_{att}** node outputs to a multiplier node ($*$) which also receives input from a gain block K_3 . The output of this multiplier is added to the output of a **CALCULATE SPECULAR 2** node (which also receives input from an **IMAGE SEQUENTIAL READ ITERATOR**) at an adder node. The output of this adder is multiplied by the output of the **CALCULATE f_{att}** node at another multiplier node ($*$). The output of this second multiplier is added to the output of a **CALCULATE AMBIENT** node (which also receives input from the **IMAGE SEQUENTIAL READ ITERATOR**) at a final adder node. The output of this final adder is fed back to the **CALCULATE N** node and also feeds into the **OUT-IMAGE SEQUENTIAL WRITE ITERATOR**. A dashed box encloses the final adder and the **OUT-IMAGE SEQUENTIAL READ ITERATOR**, which feeds back into the final adder. A legend at the bottom right indicates that the dashed box represents **2ND AND SUBSEQUENT LIGHTS**.

FIG. 150

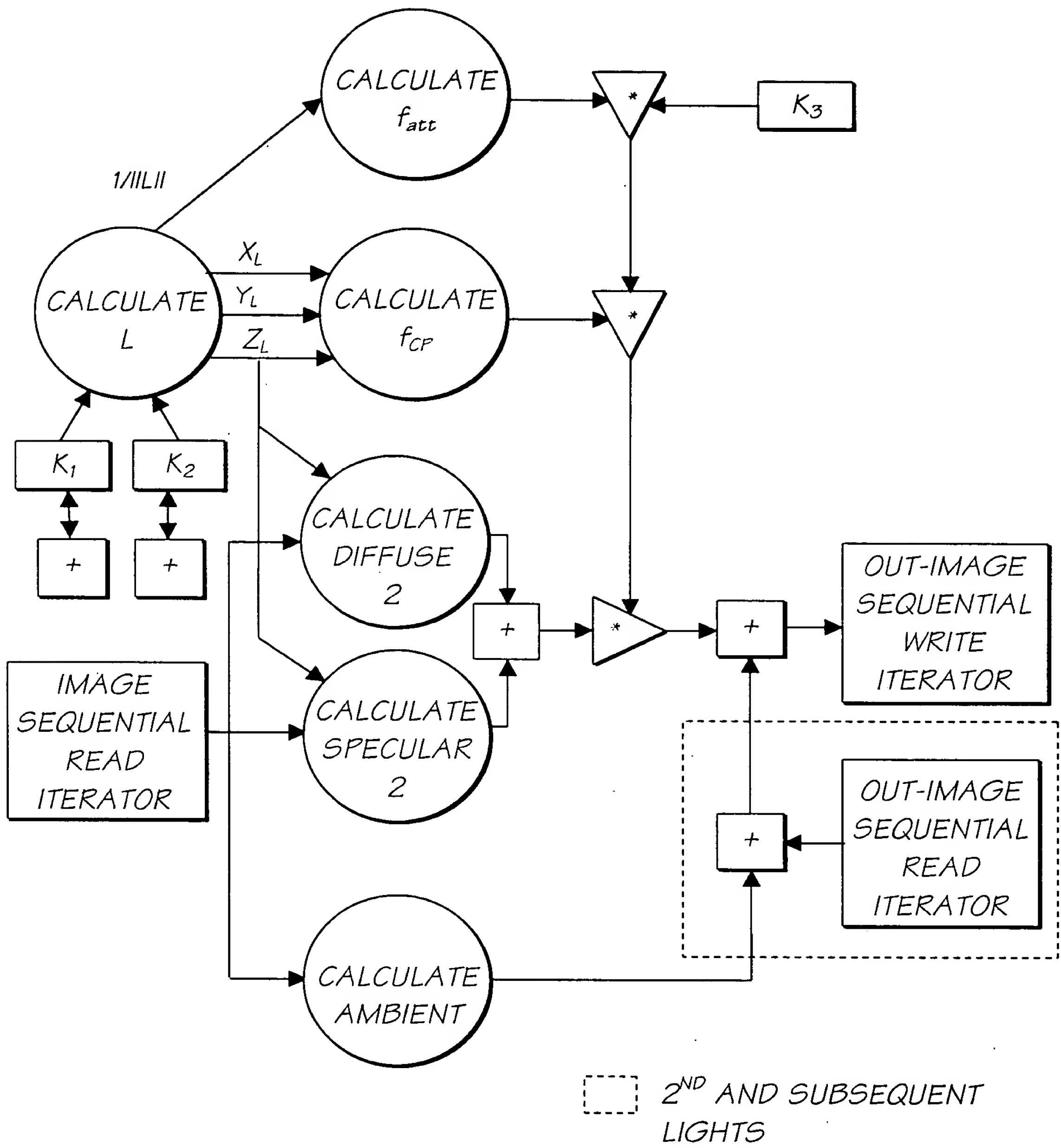


FIG. 151

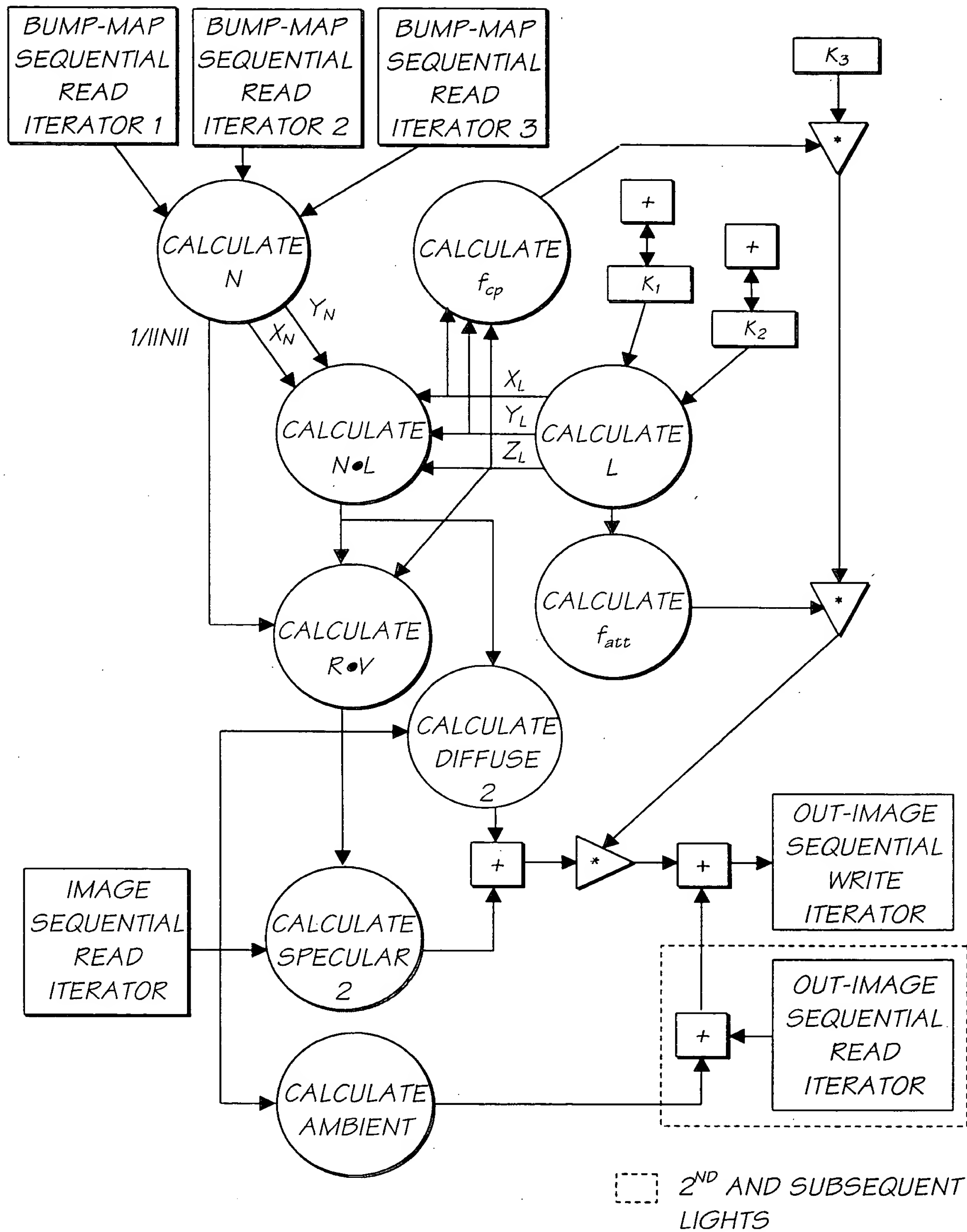
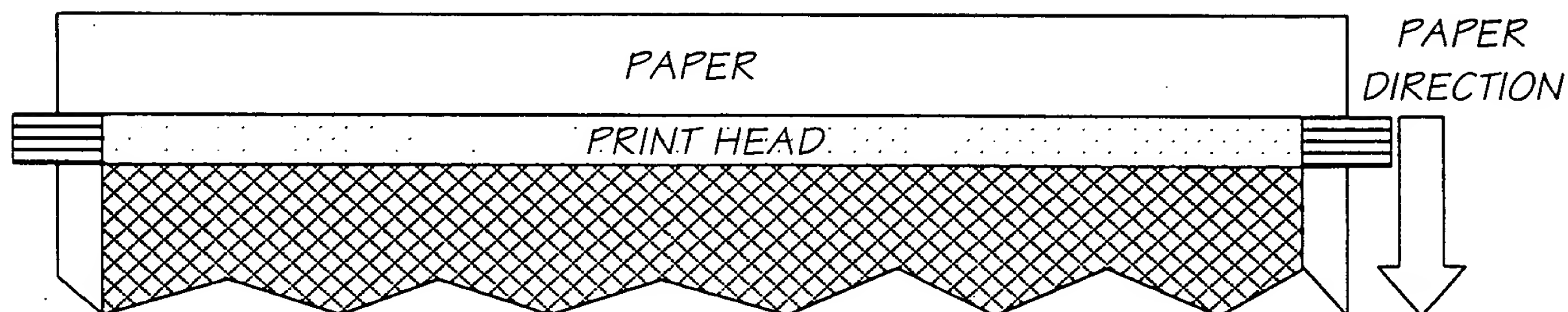


FIG. 152



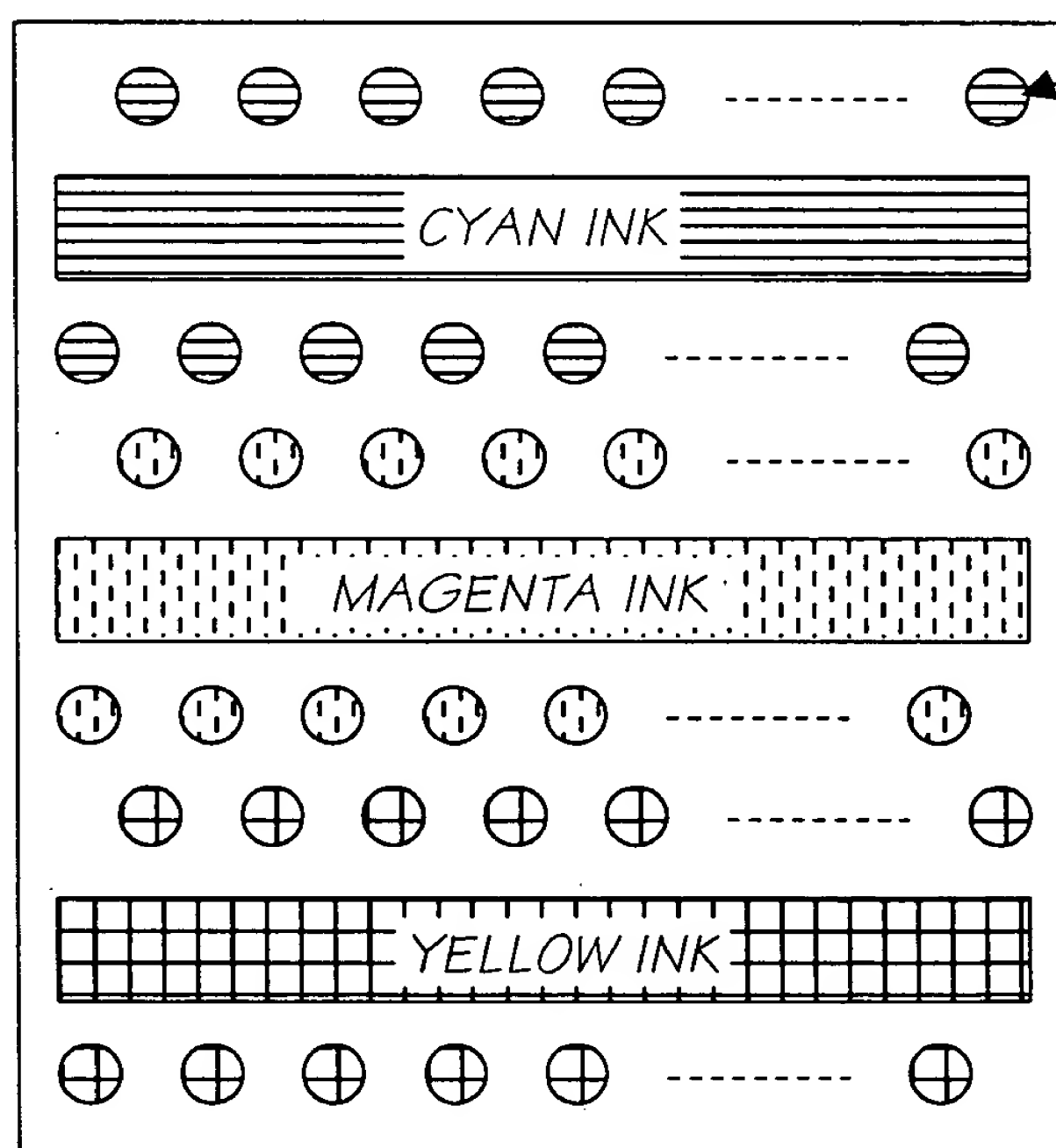
8 PRINT HEAD SEGMENTS IN PRINT HEAD

SEGMENT	SEGMENT	SEGMENT	SEGMENT	SEGMENT	SEGMENT	SEGMENT	SEGMENT
0	1	2	3	4	5	6	7

1250 μM (375 DOTS PER SEGMENT ROW,
OR 750 DOTS PER SEGMENT COLOR)

1 DOT IS 16.6 μM IN
DIAMETER

(A 100 μM SQUARE =
6 X 6 = 36 DOTS)



466.6 μM
(28 DOTS)

33.3 μM
(2 DOTS)

133.3 μM
(8 DOTS)

EACH SEGMENT CONTAINS 6 ROWS OF DOTS:
ODD AND EVEN CYAN, MAGENTA, AND YELLOW.

FIG. 153

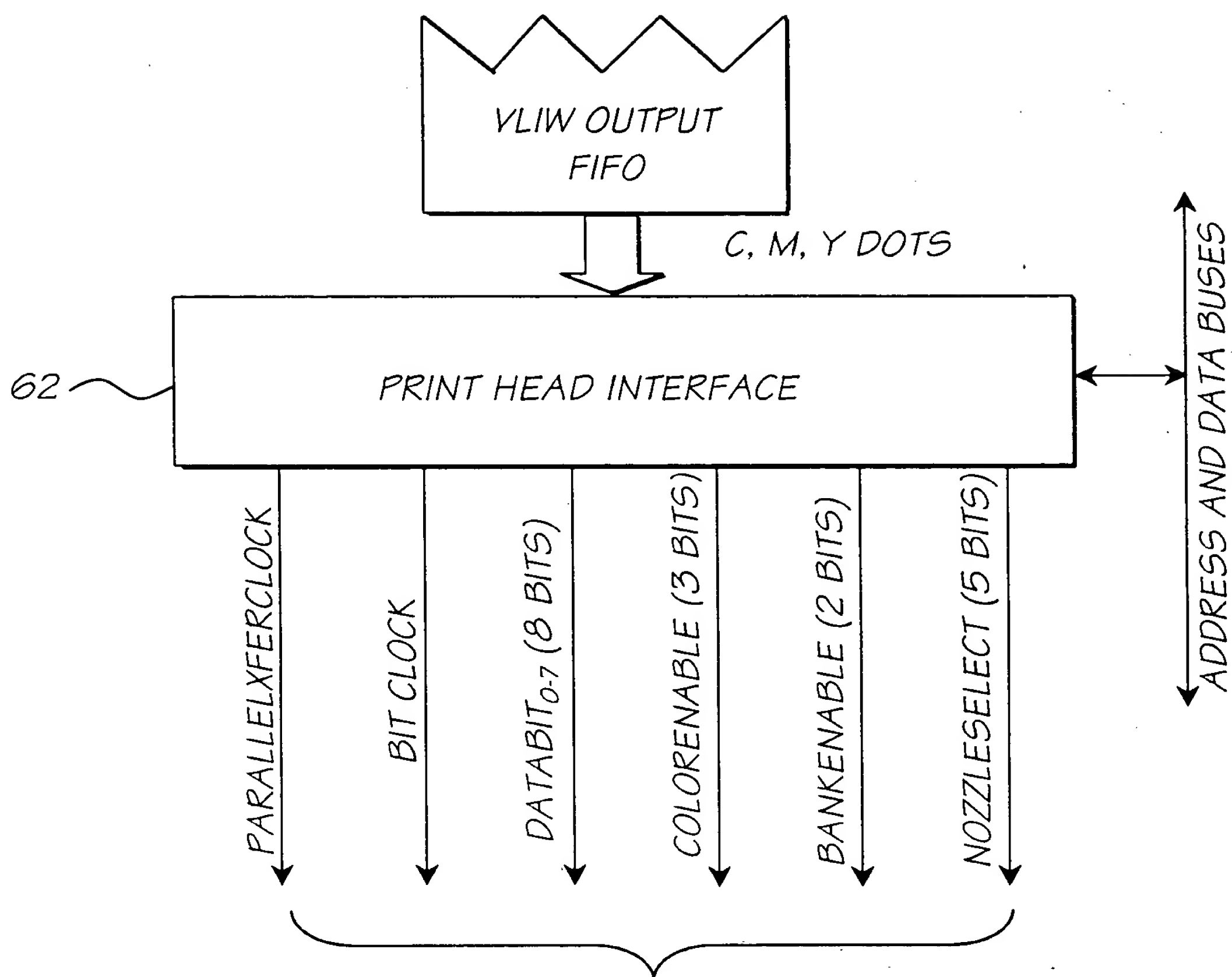


FIG. 154

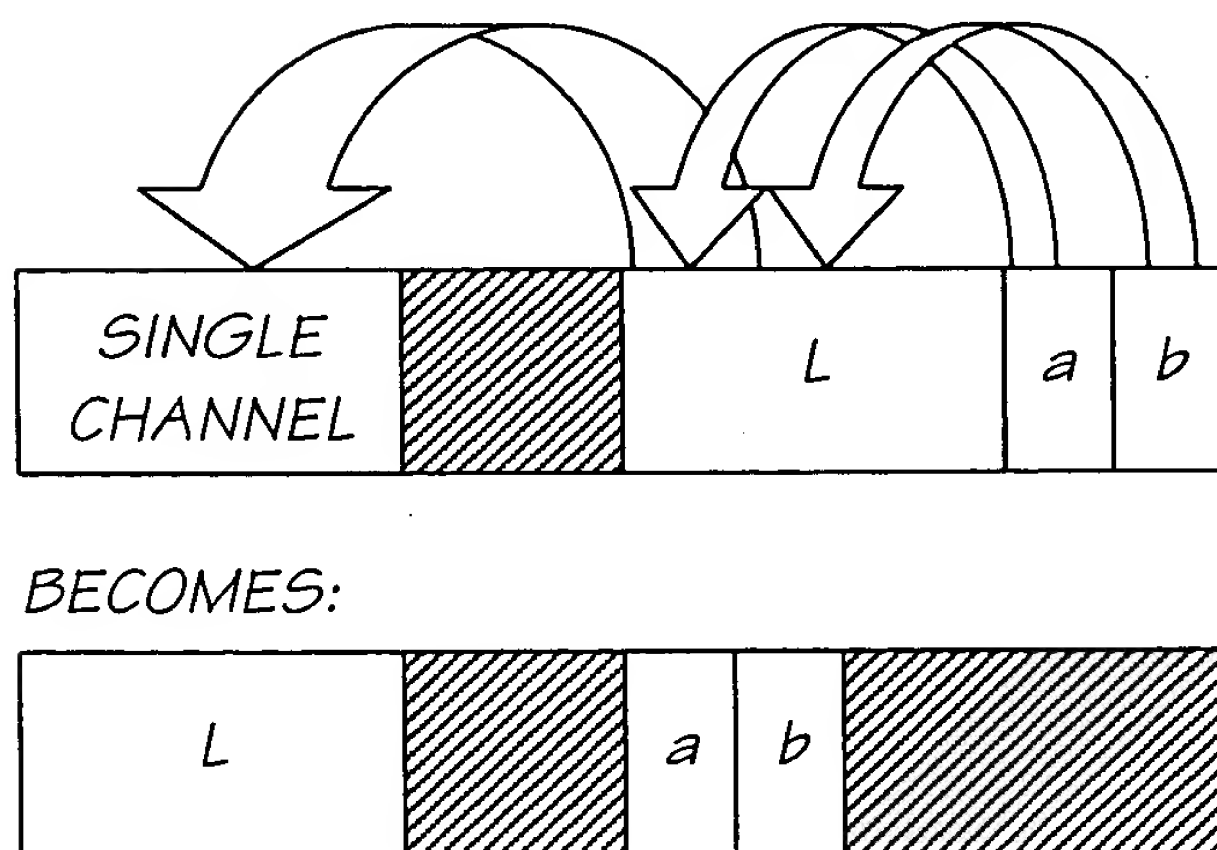


FIG. 155

FIG. 156

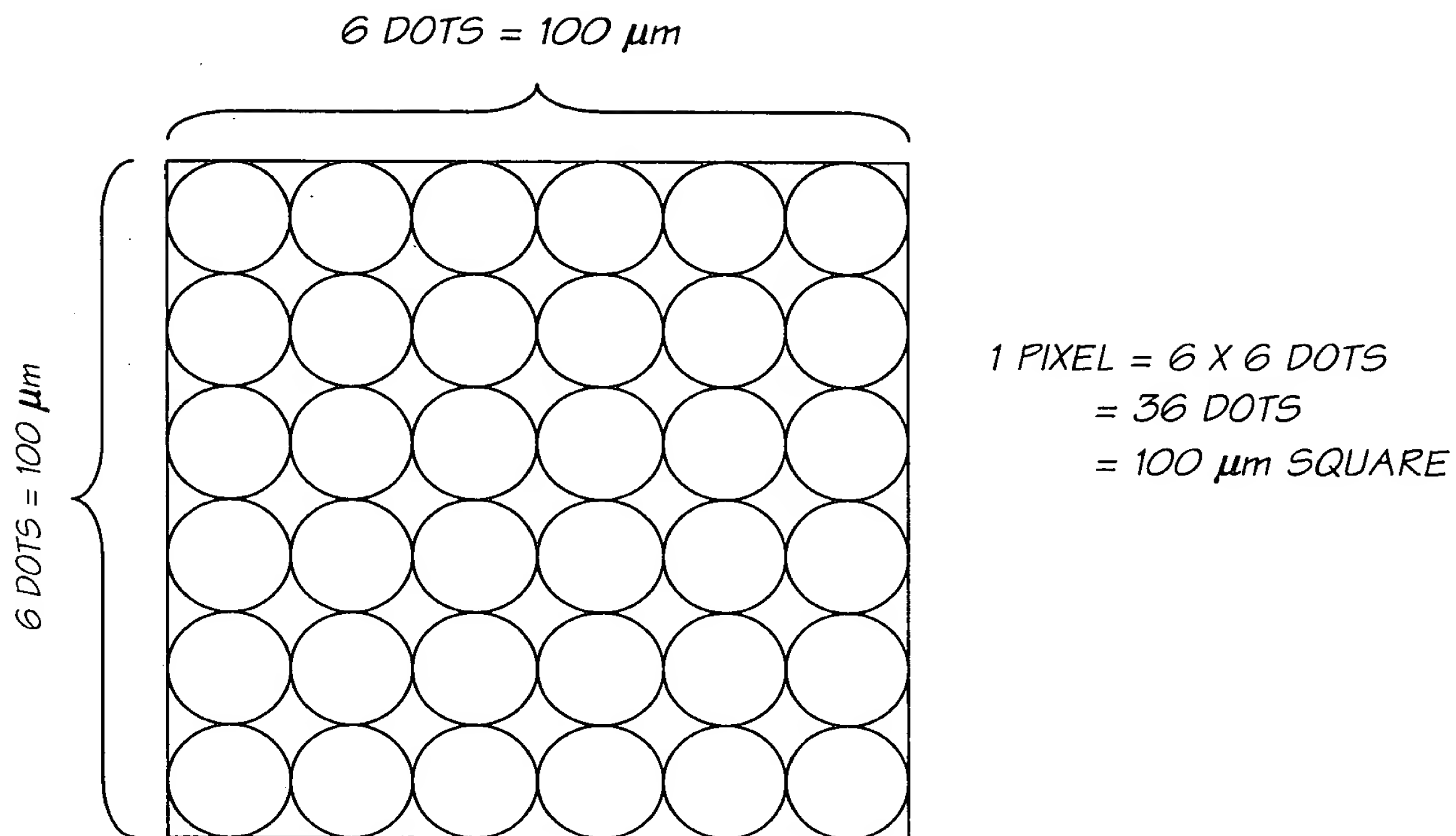


FIG. 156

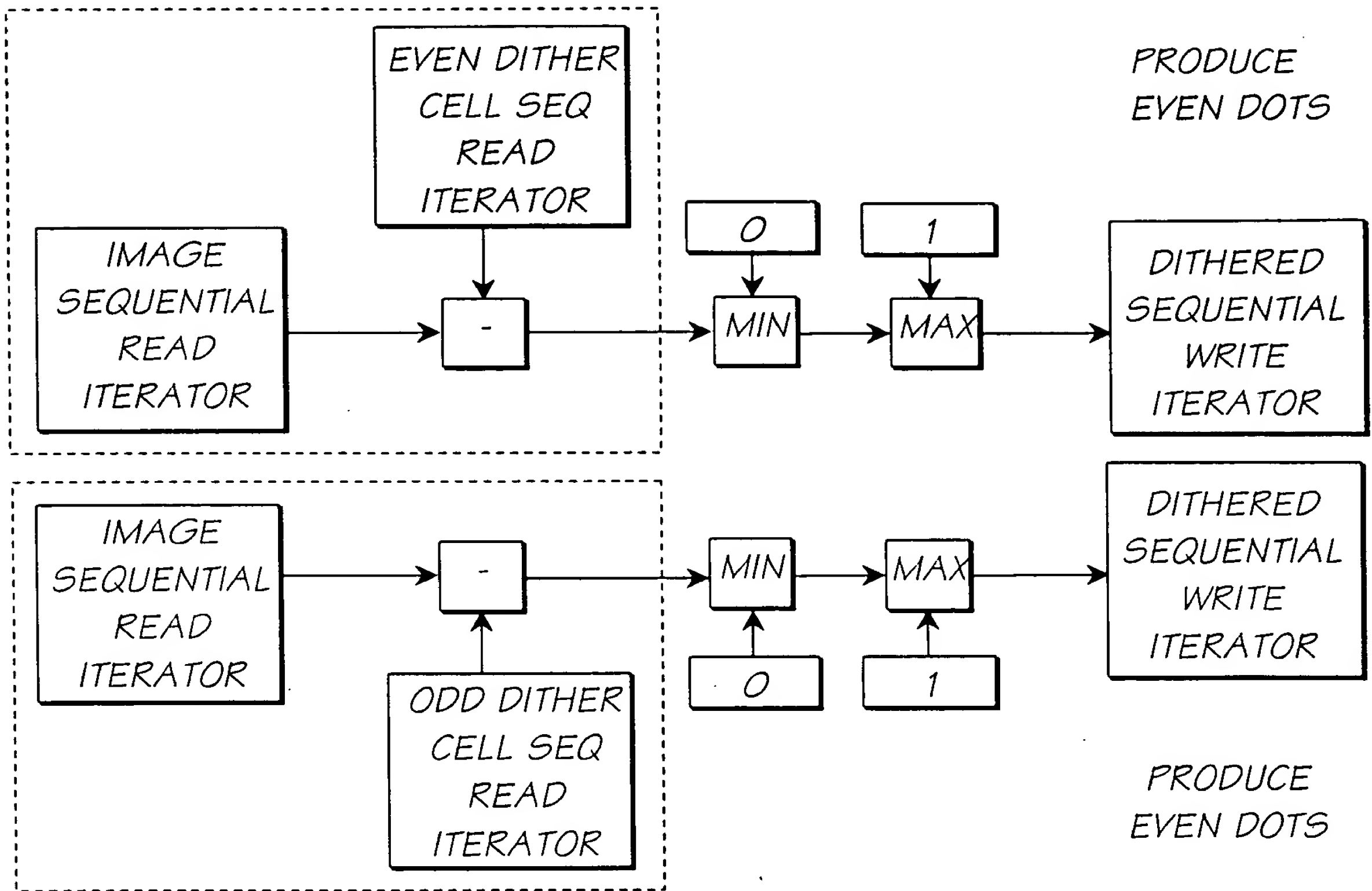


FIG. 157

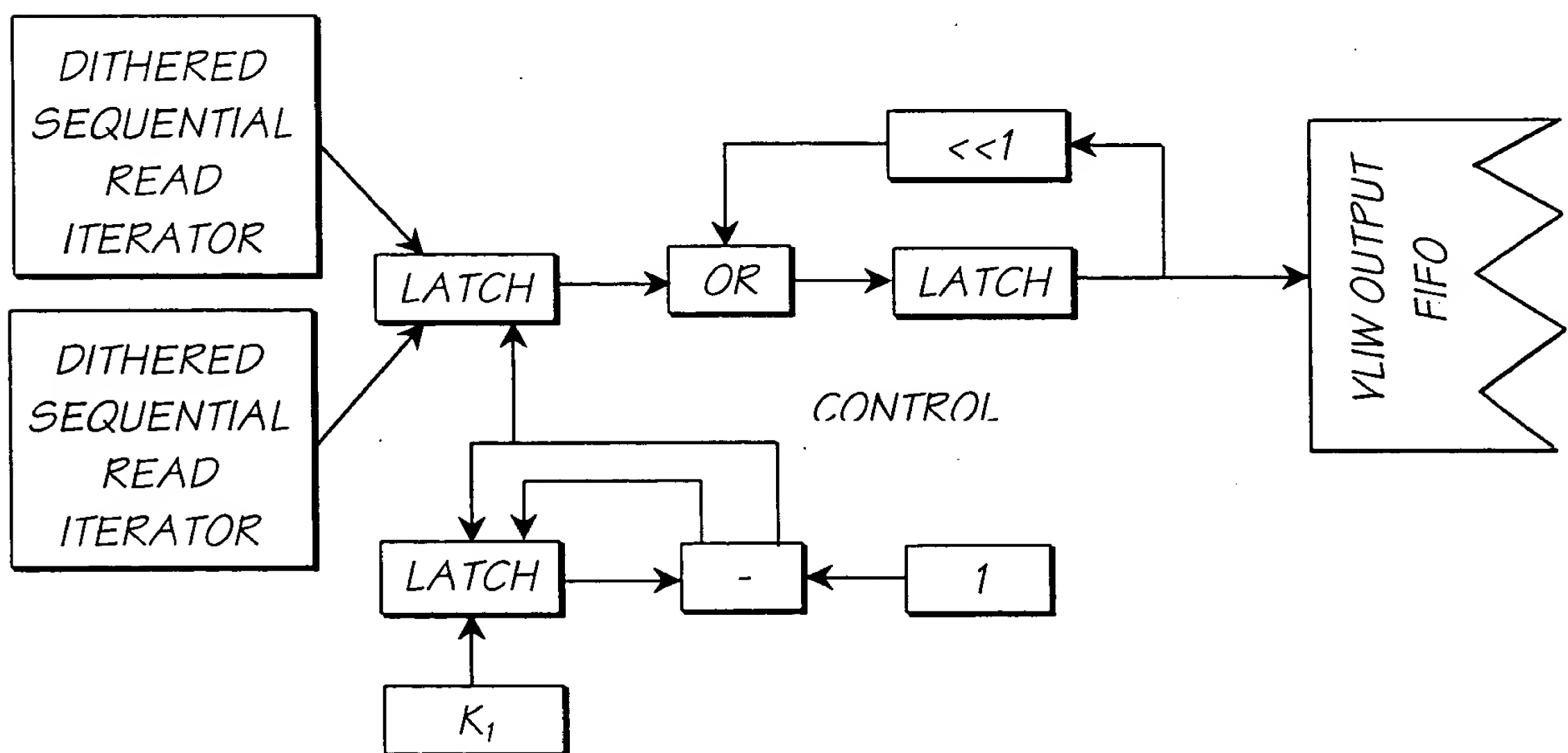


FIG. 158

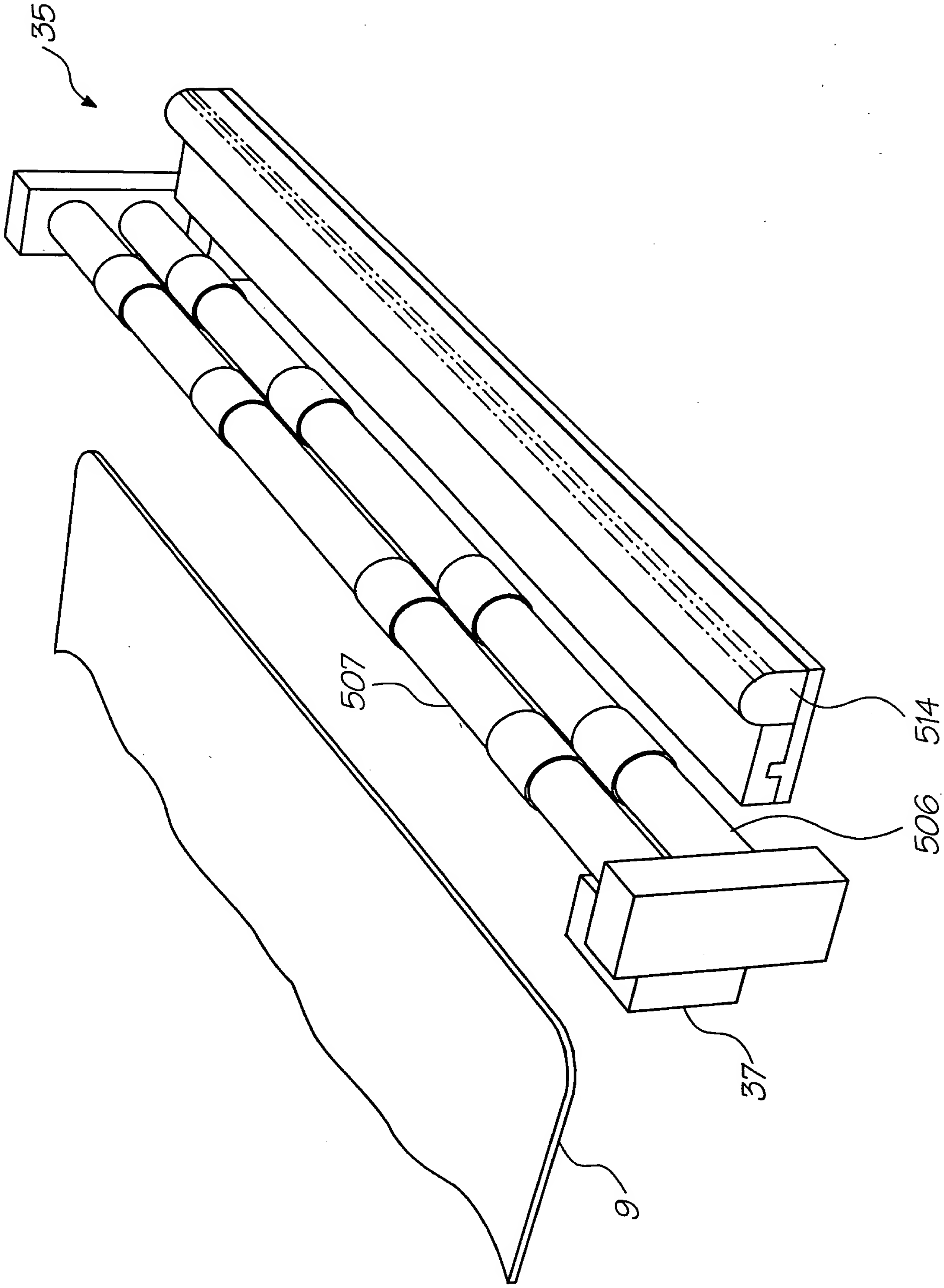


FIG. 159

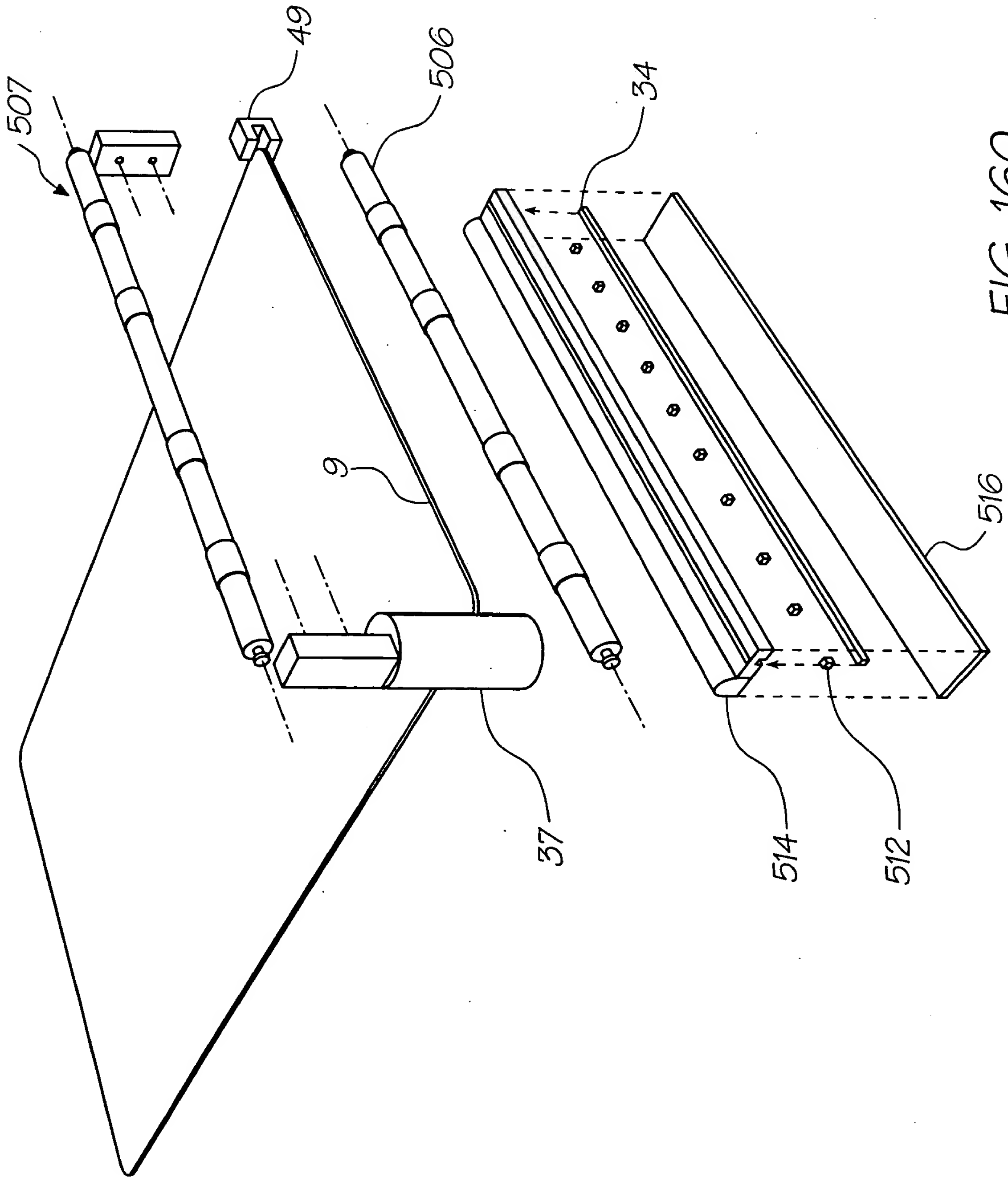
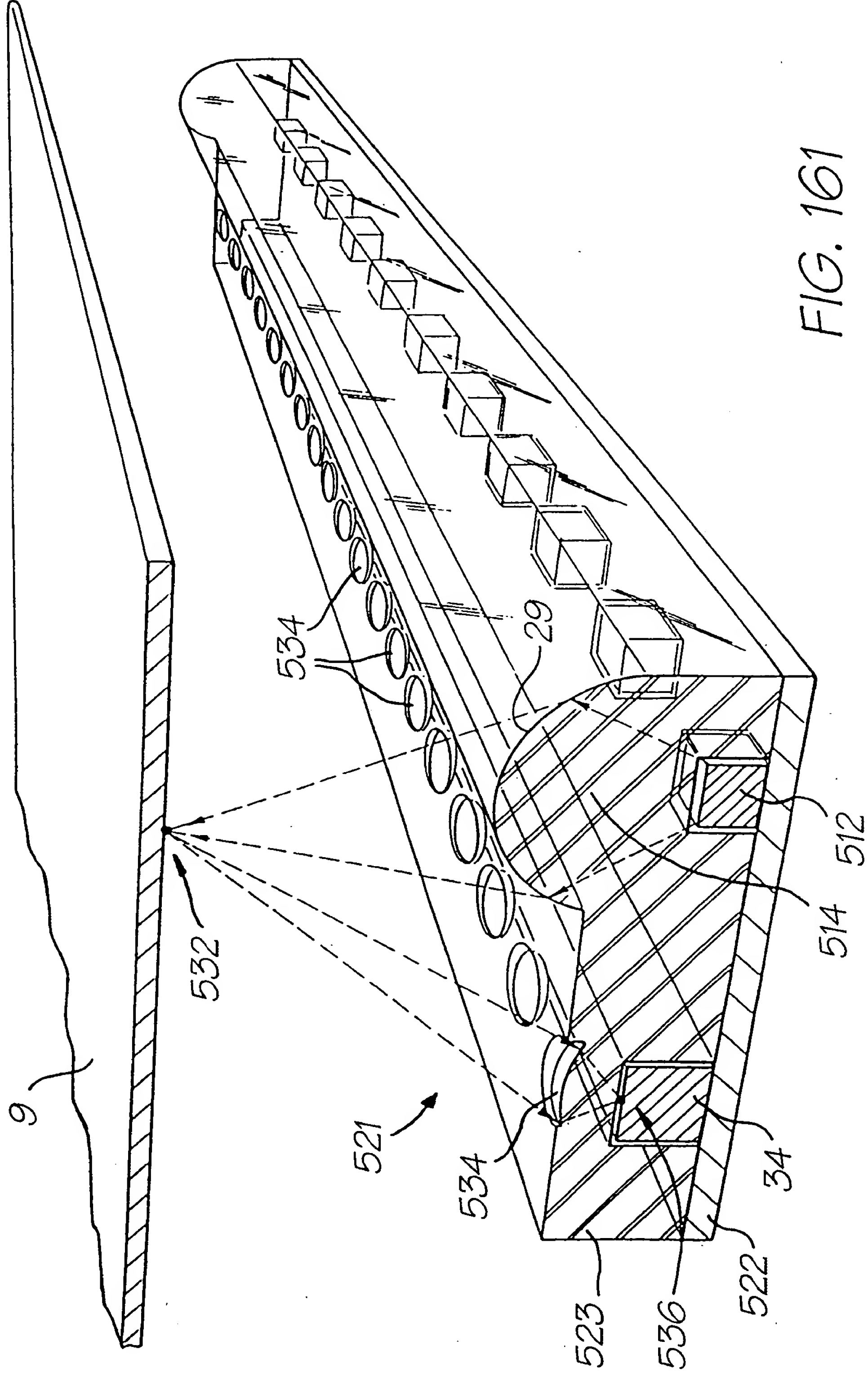


FIG. 160



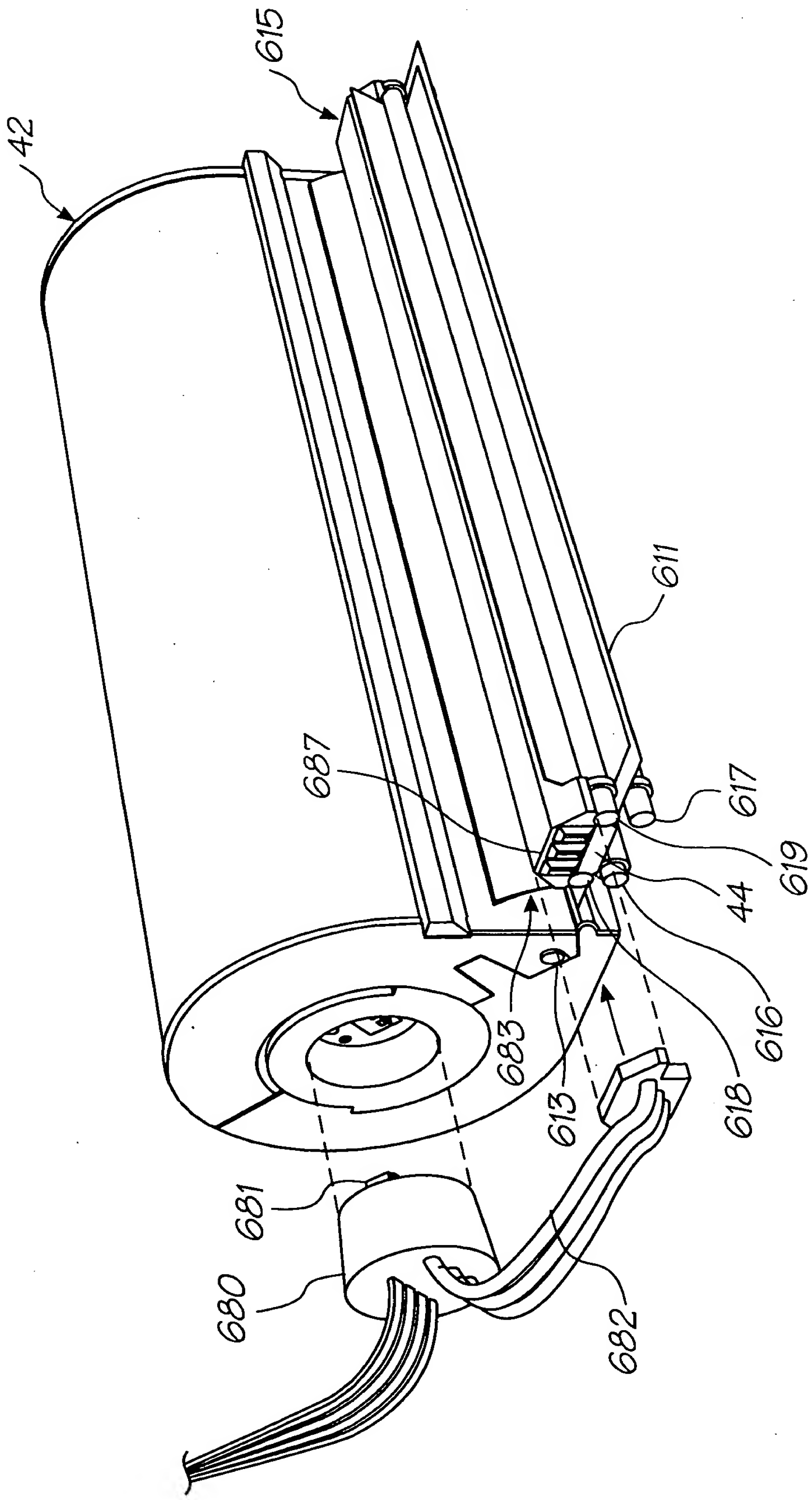


FIG. 162

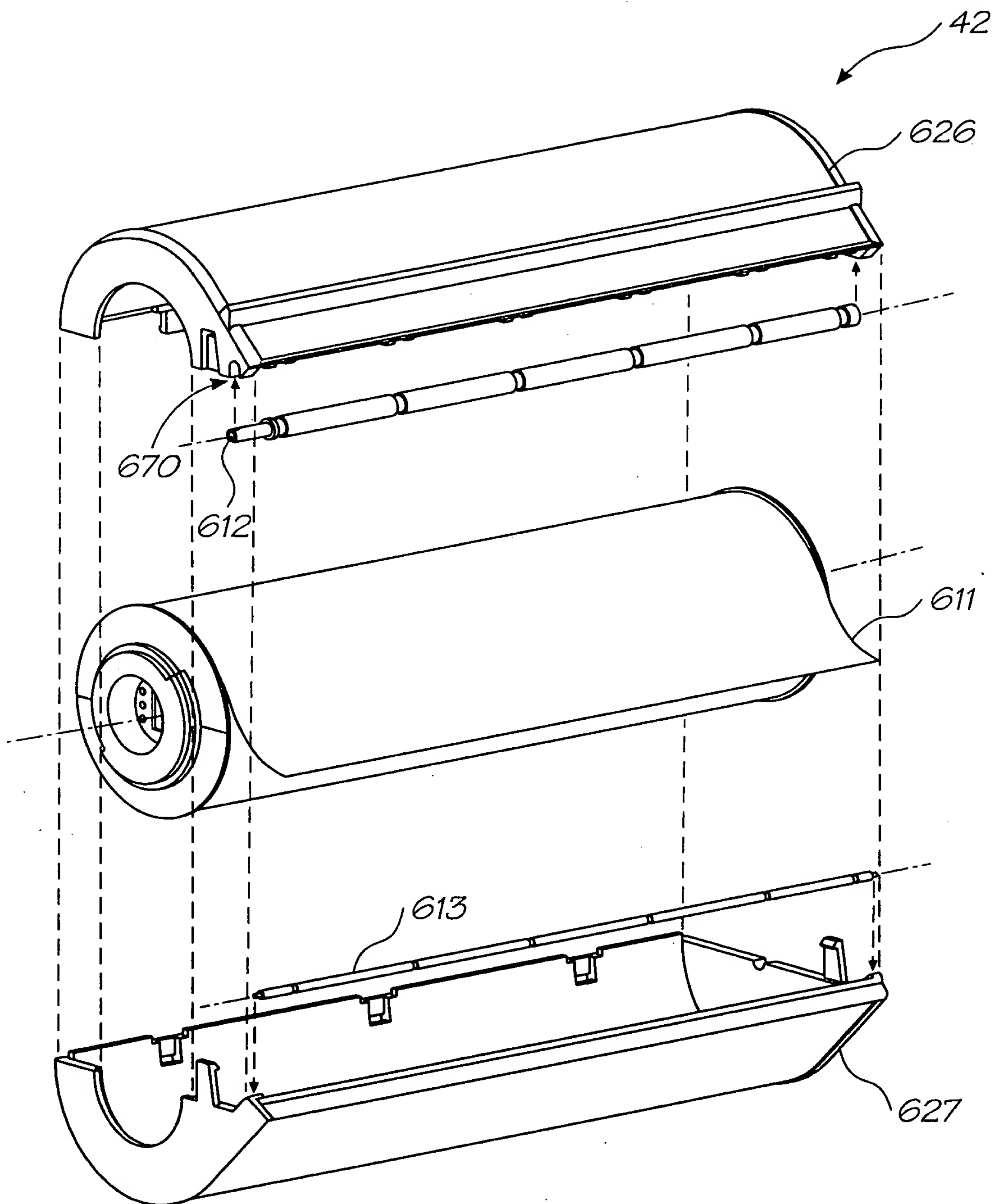


FIG. 163

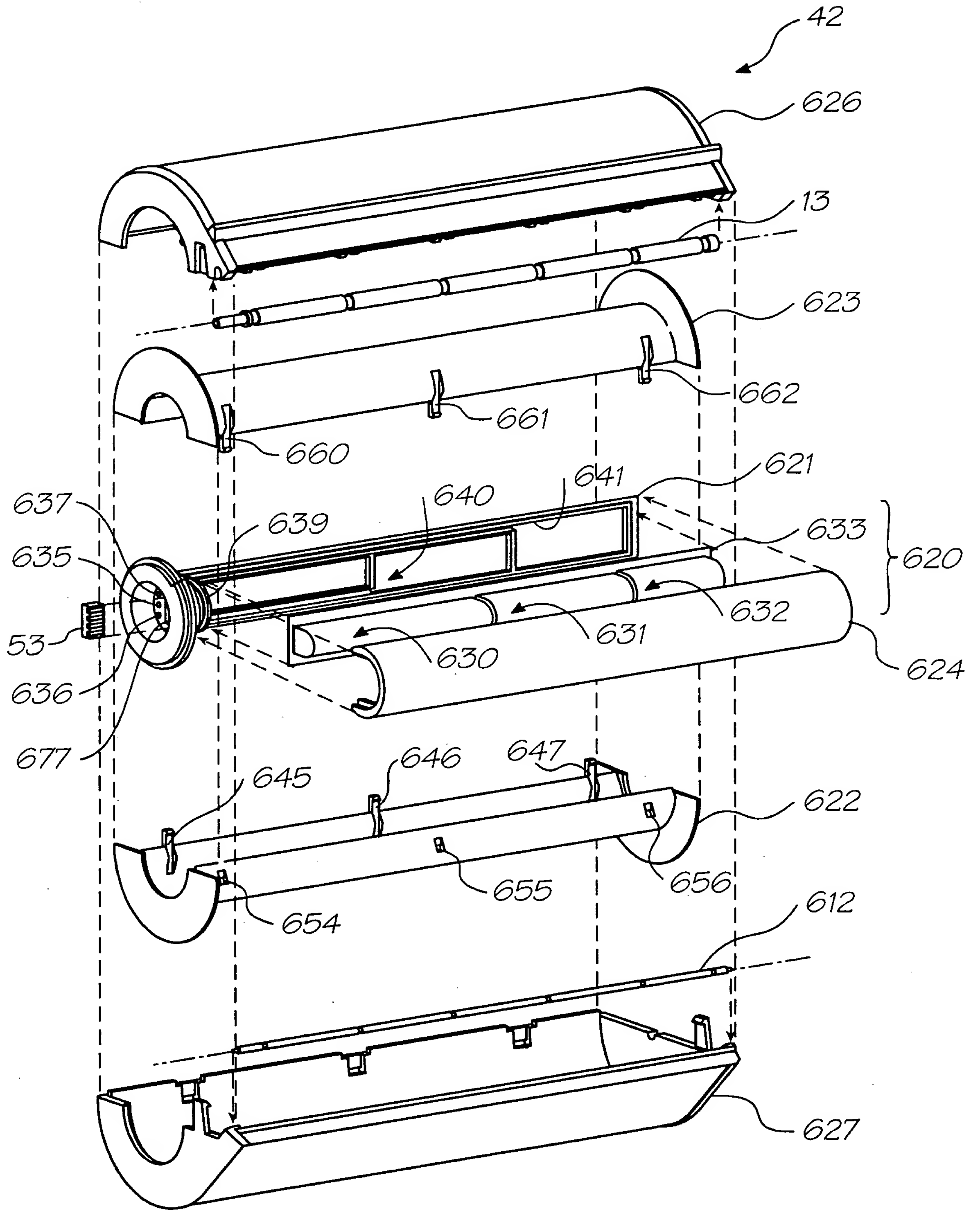


FIG. 164

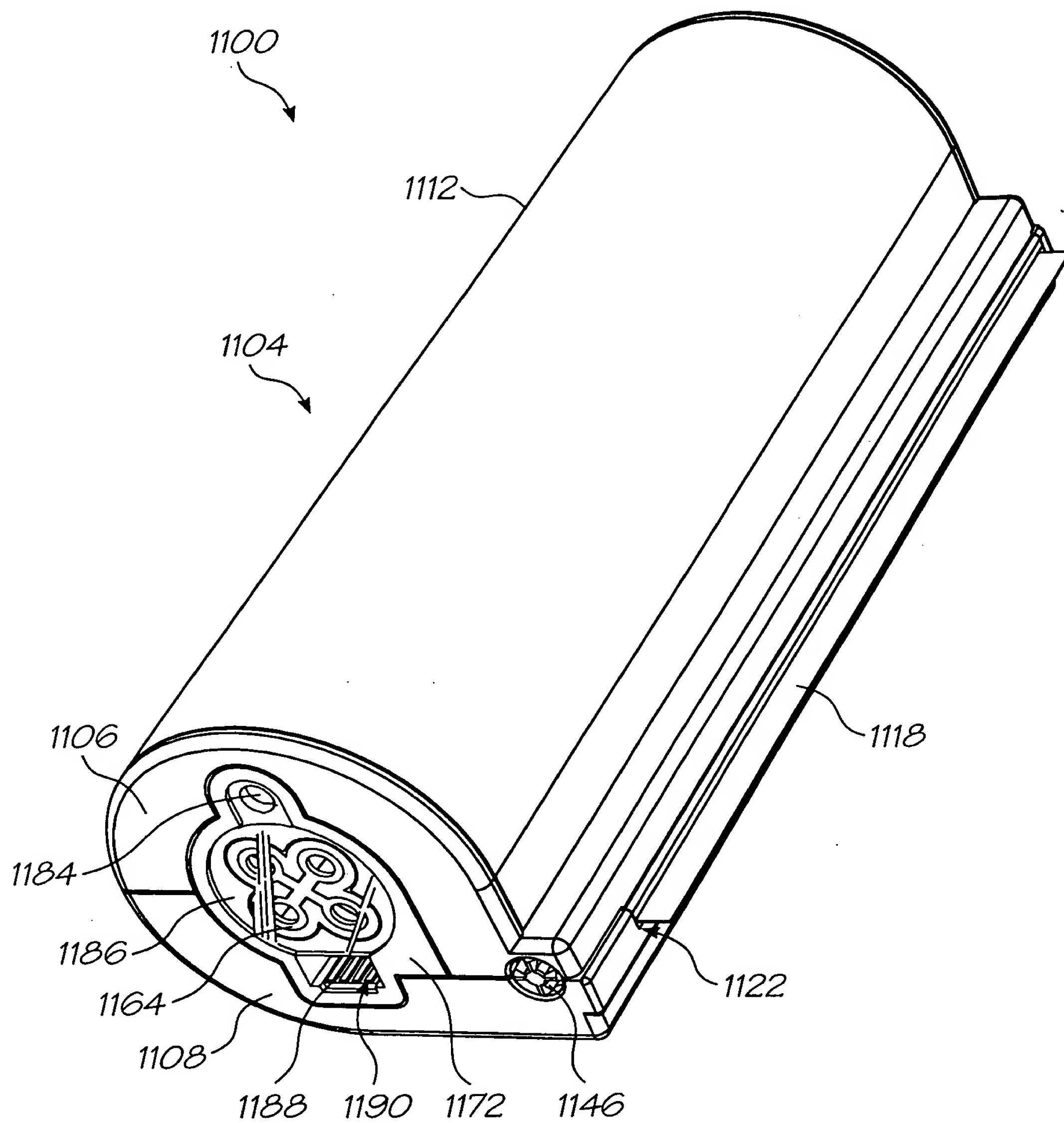


FIG. 164A

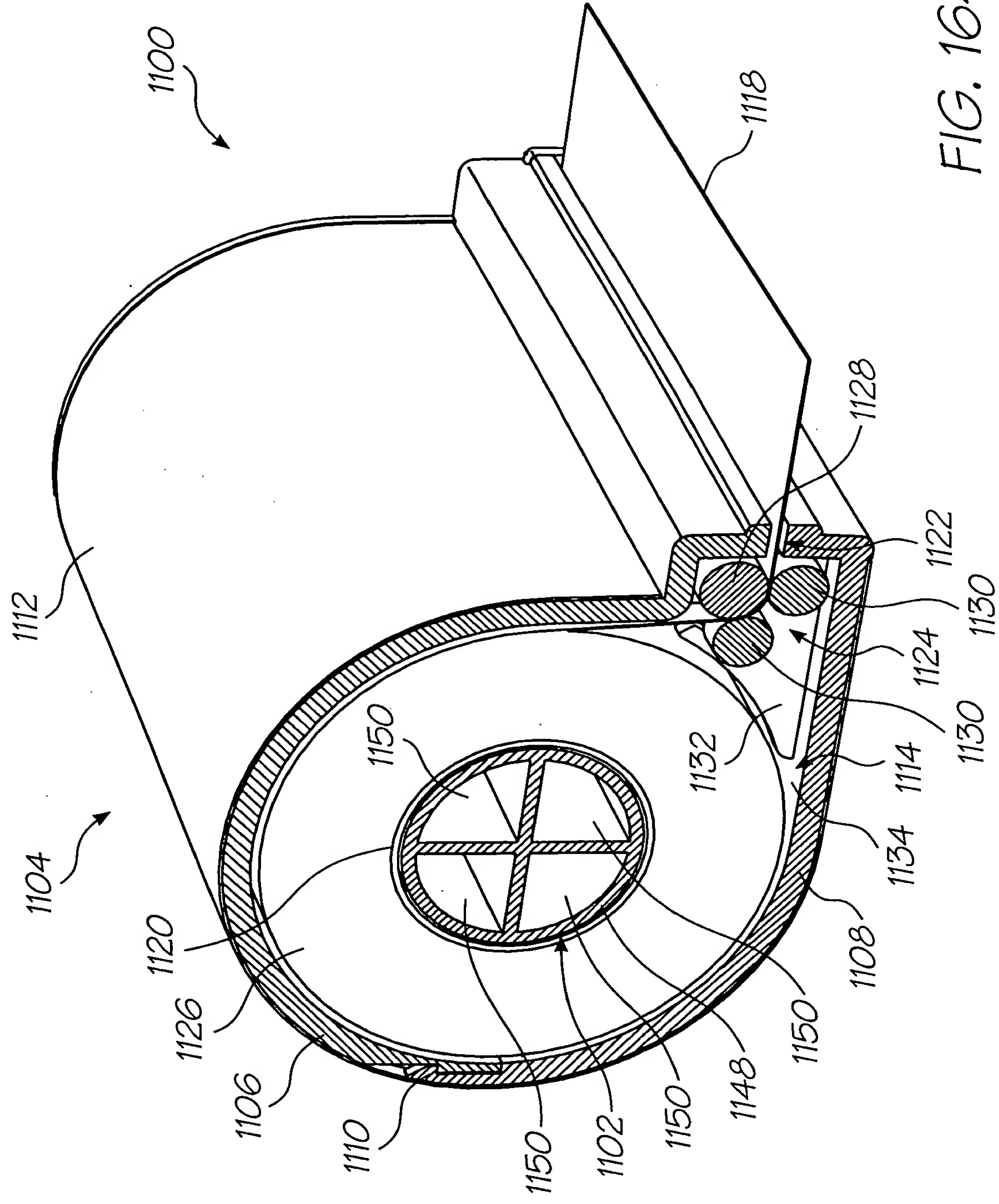


FIG. 164B

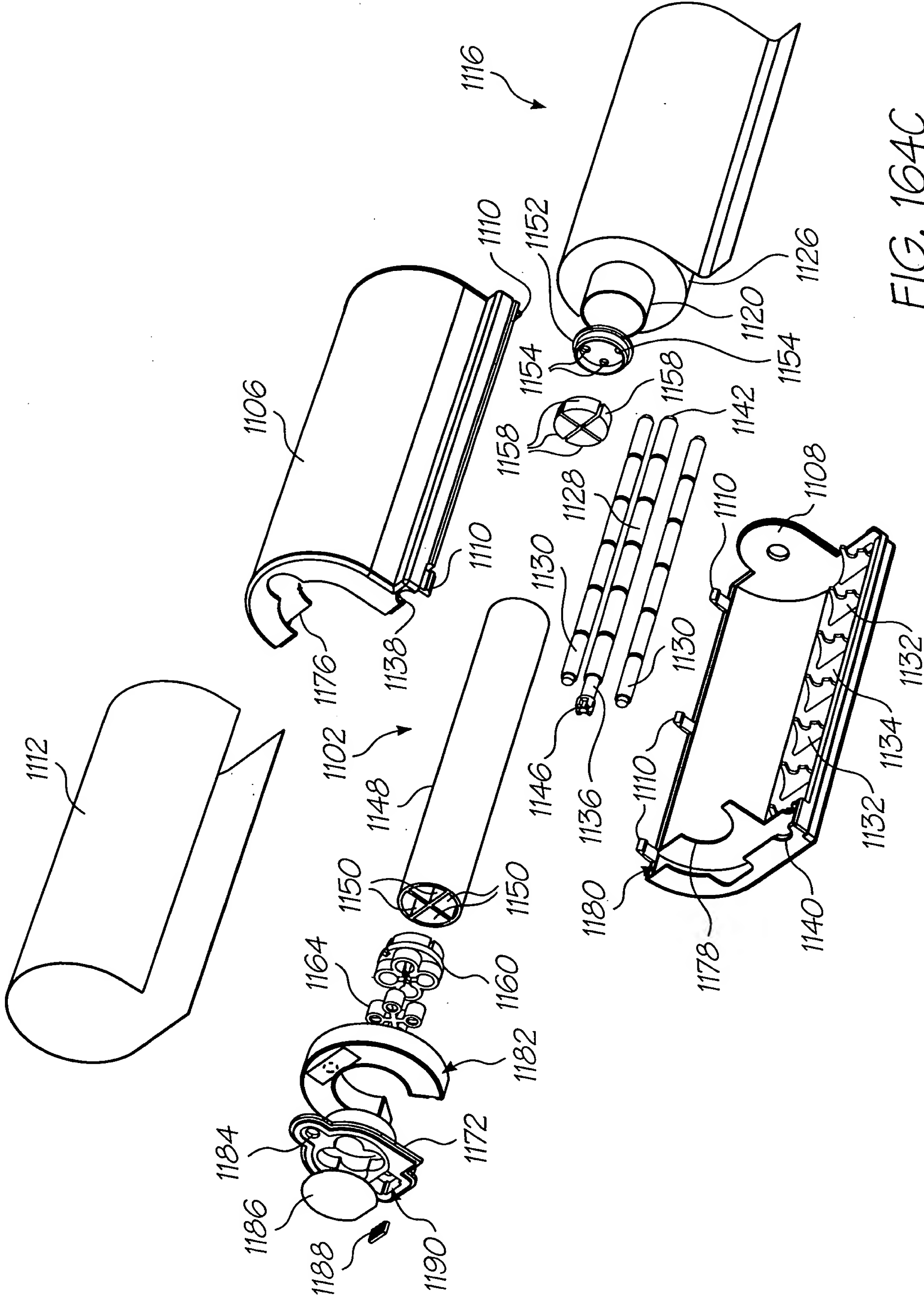


FIG. 164C

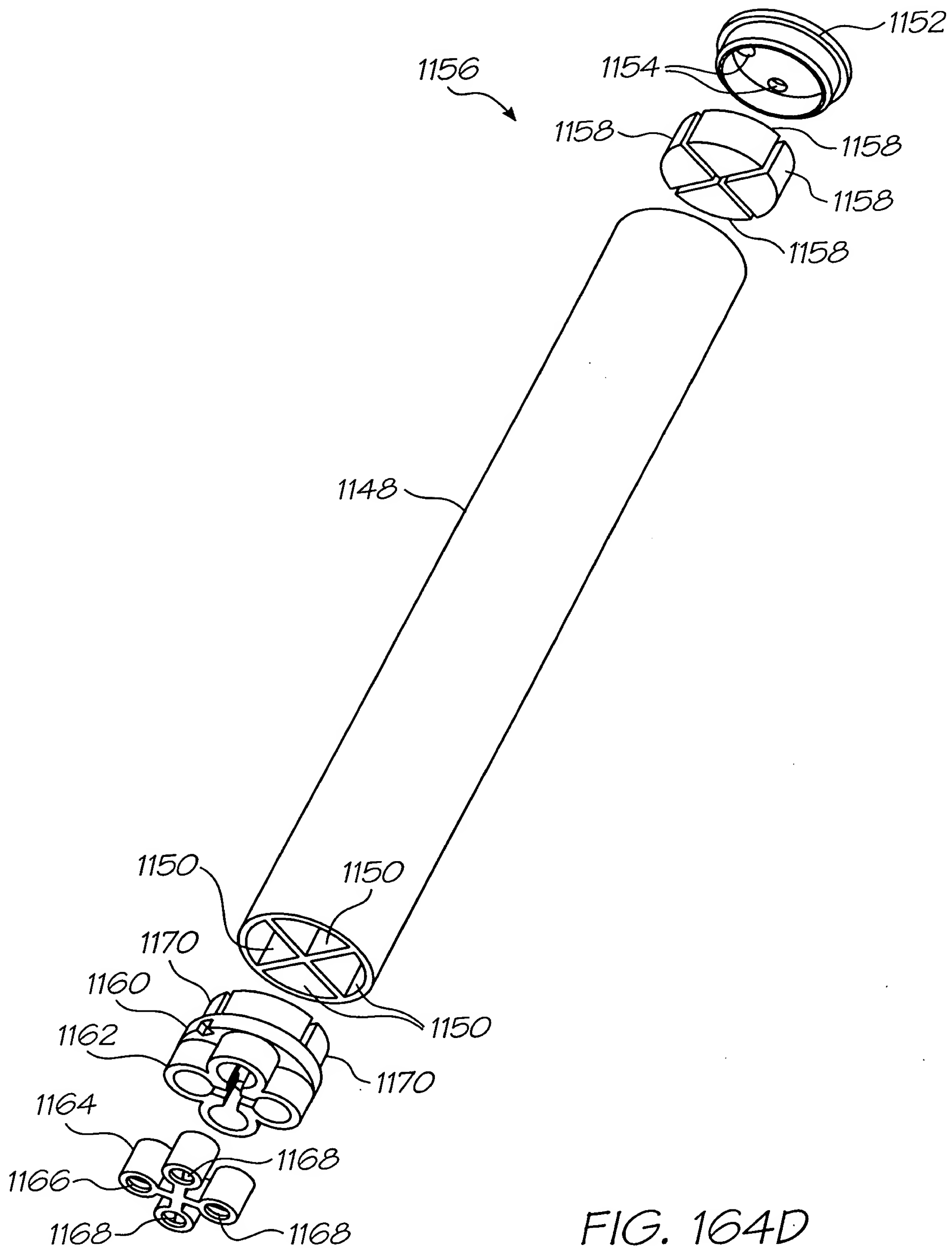


FIG. 164D

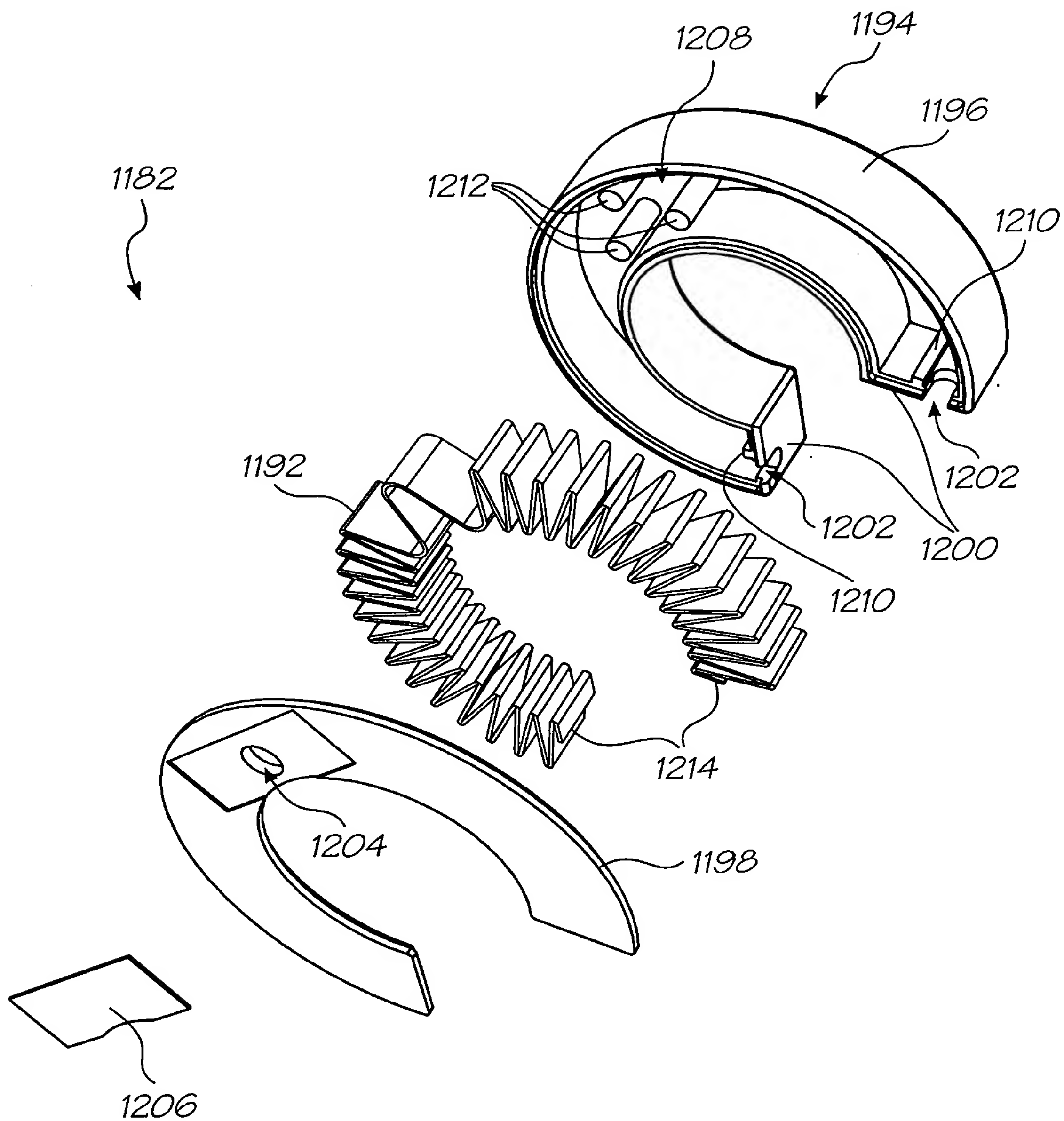


FIG. 164E

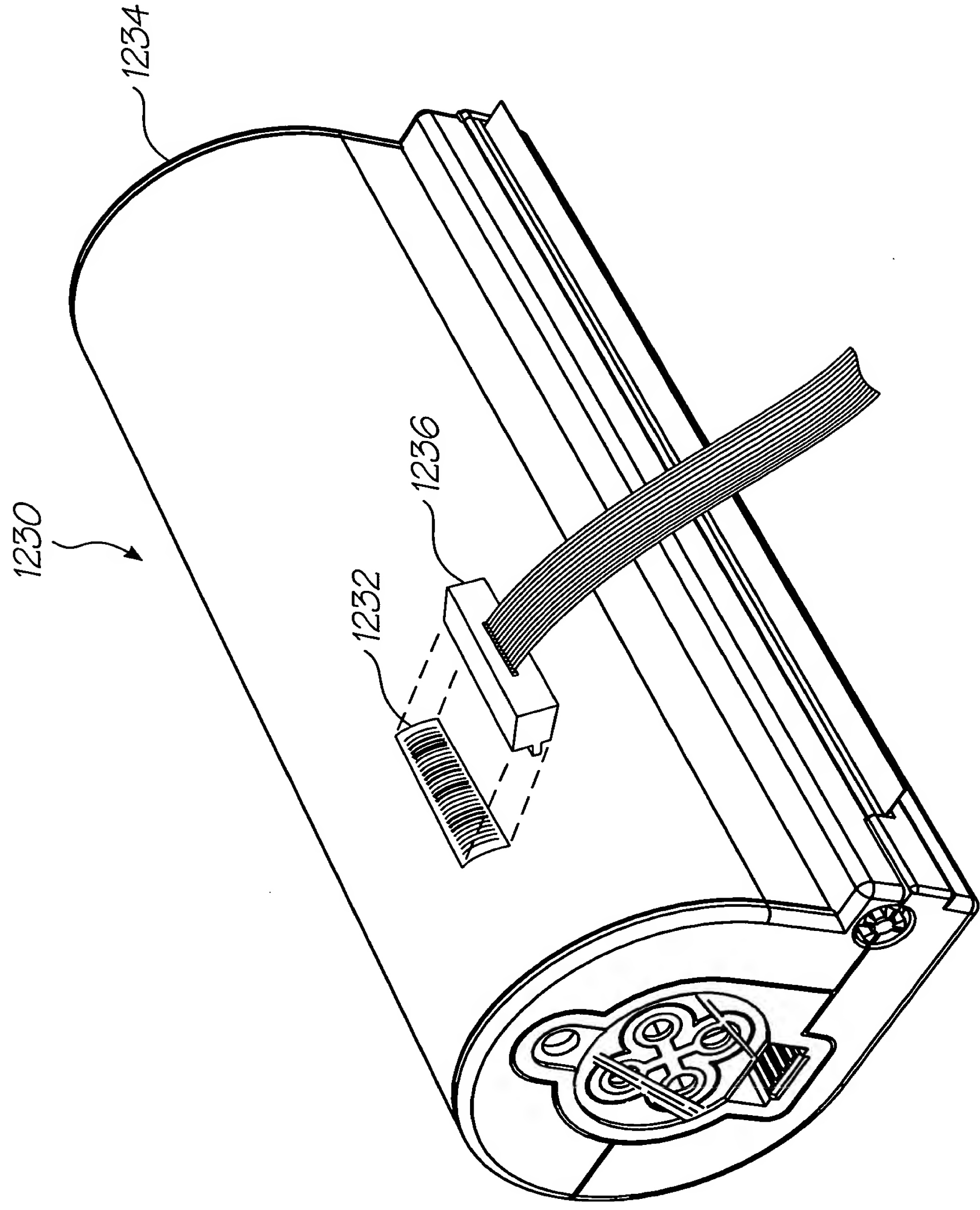


FIG. 164F

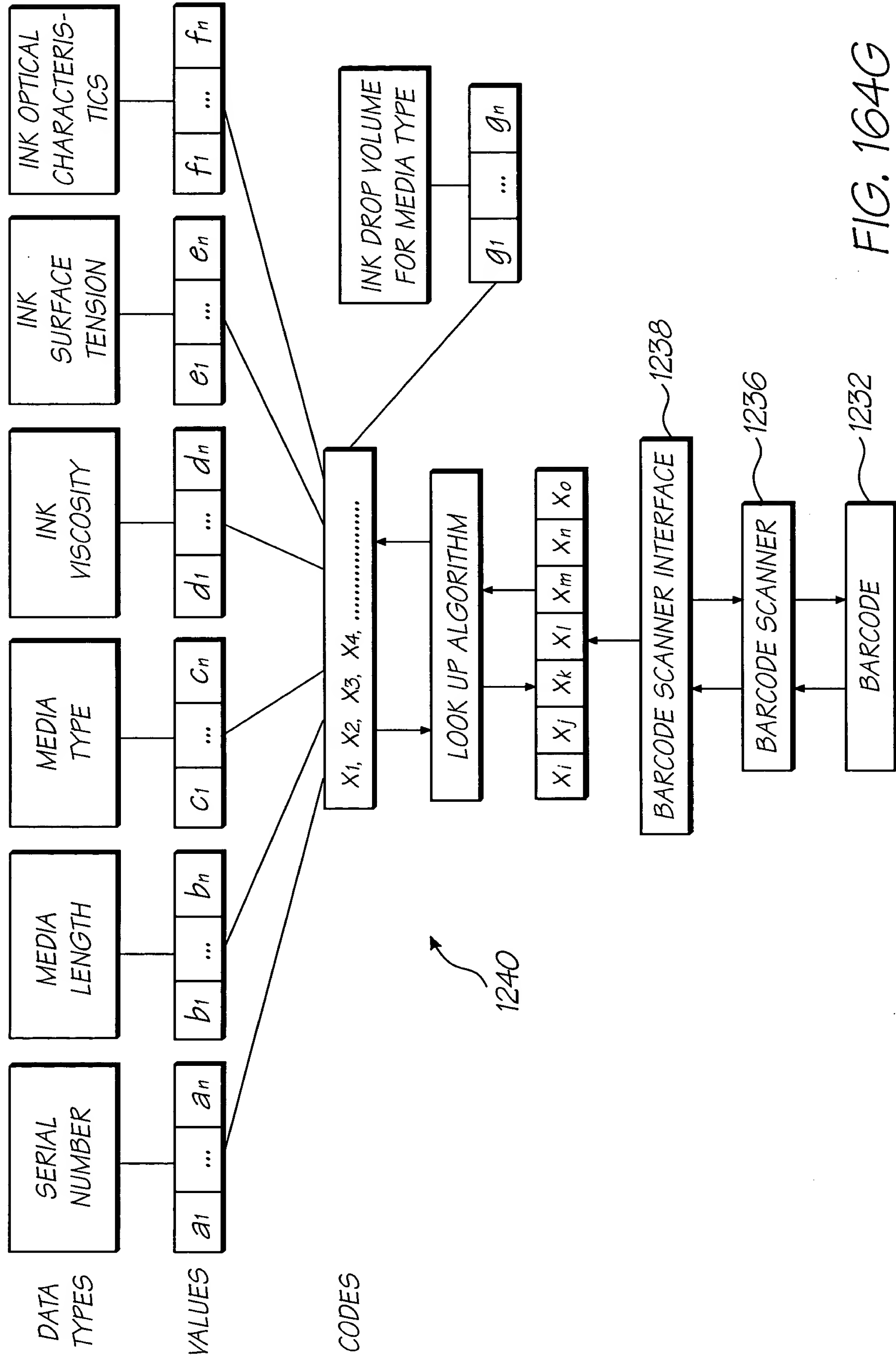


FIG. 164G

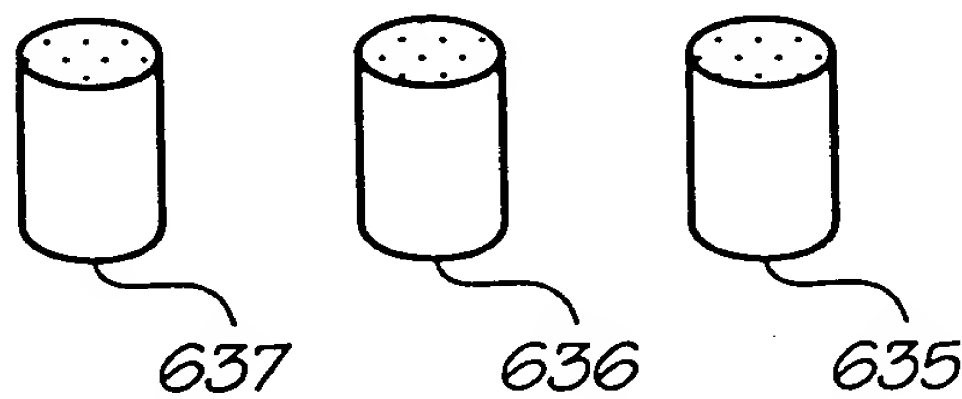
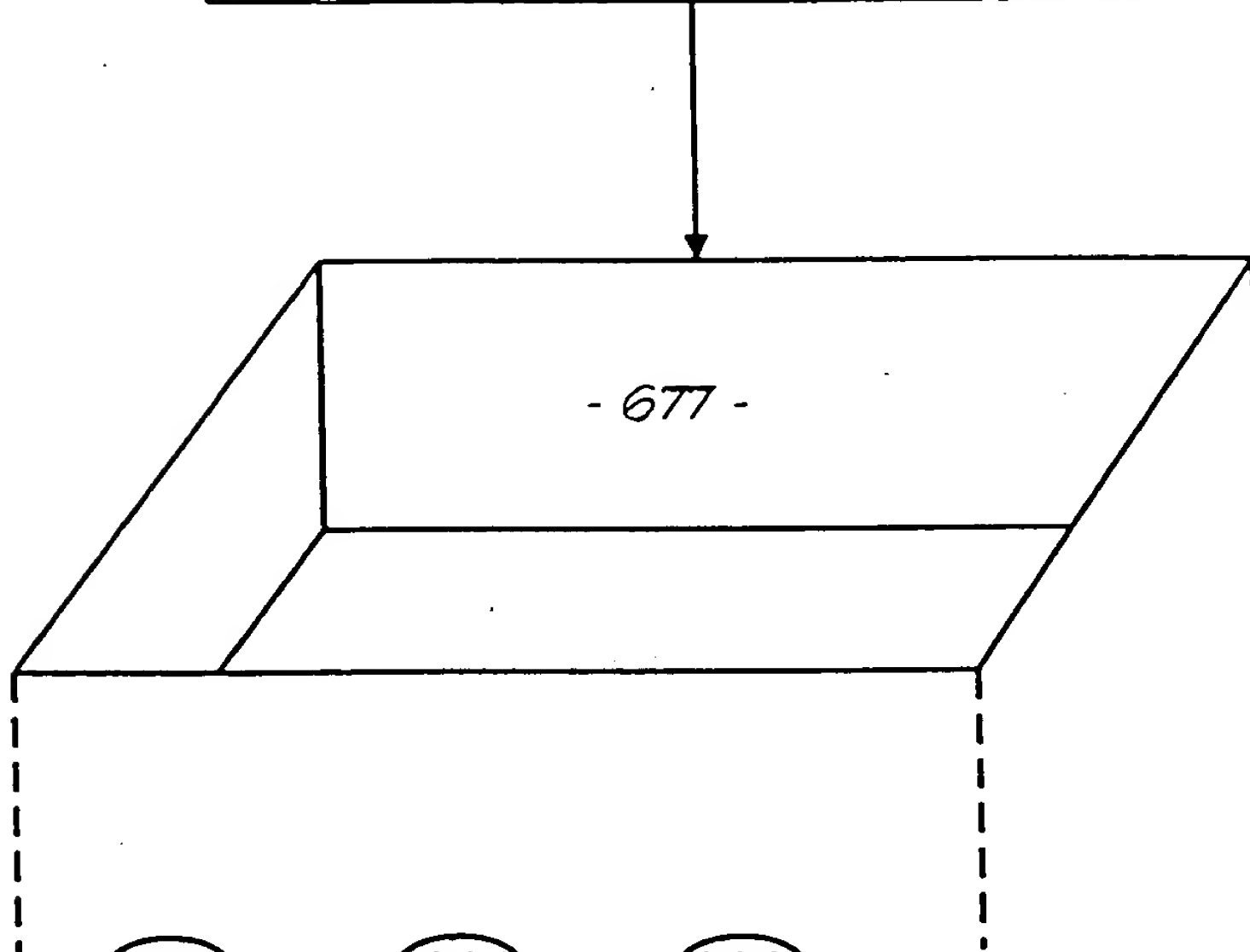
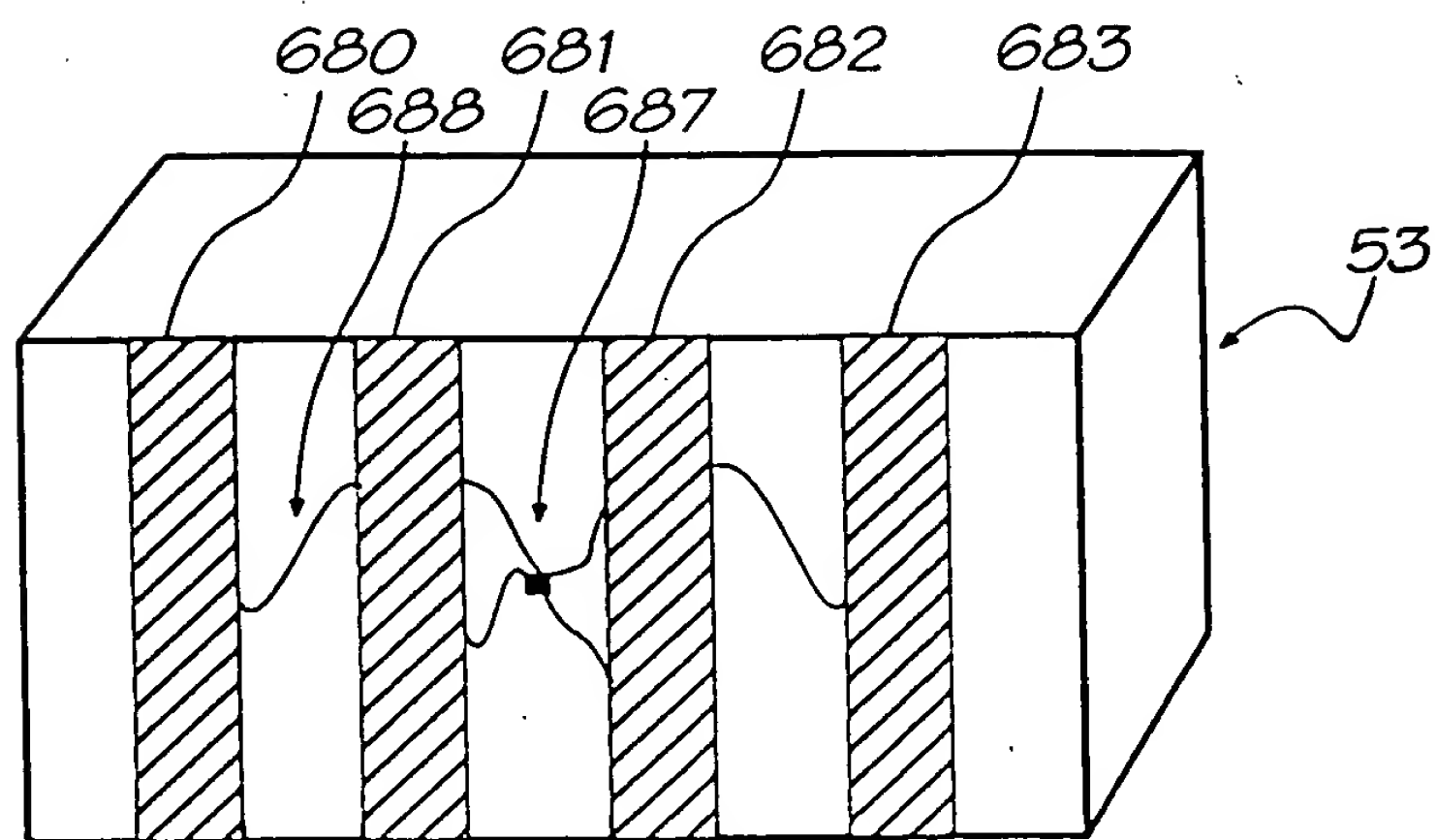


FIG. 165

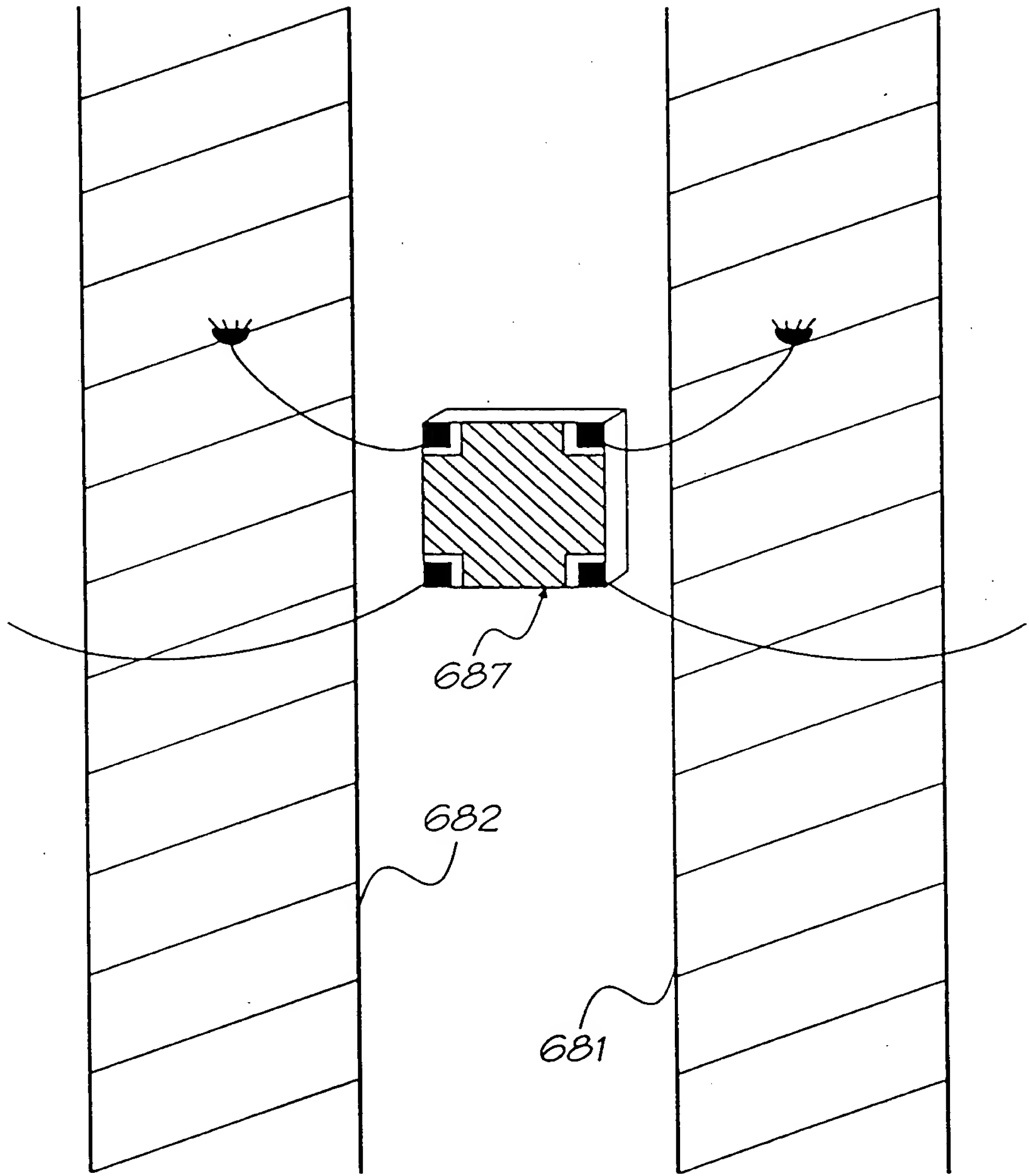


FIG. 166

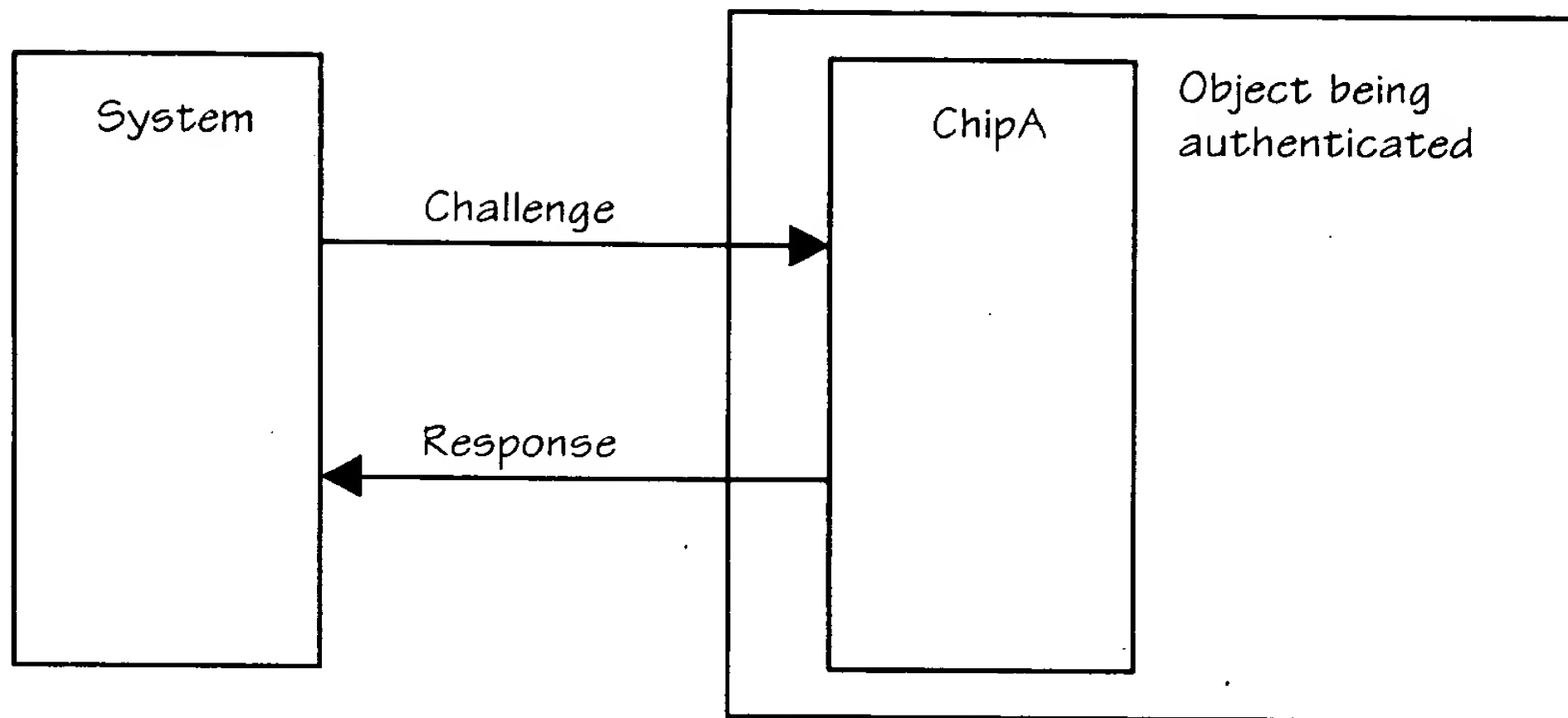


FIG. 167

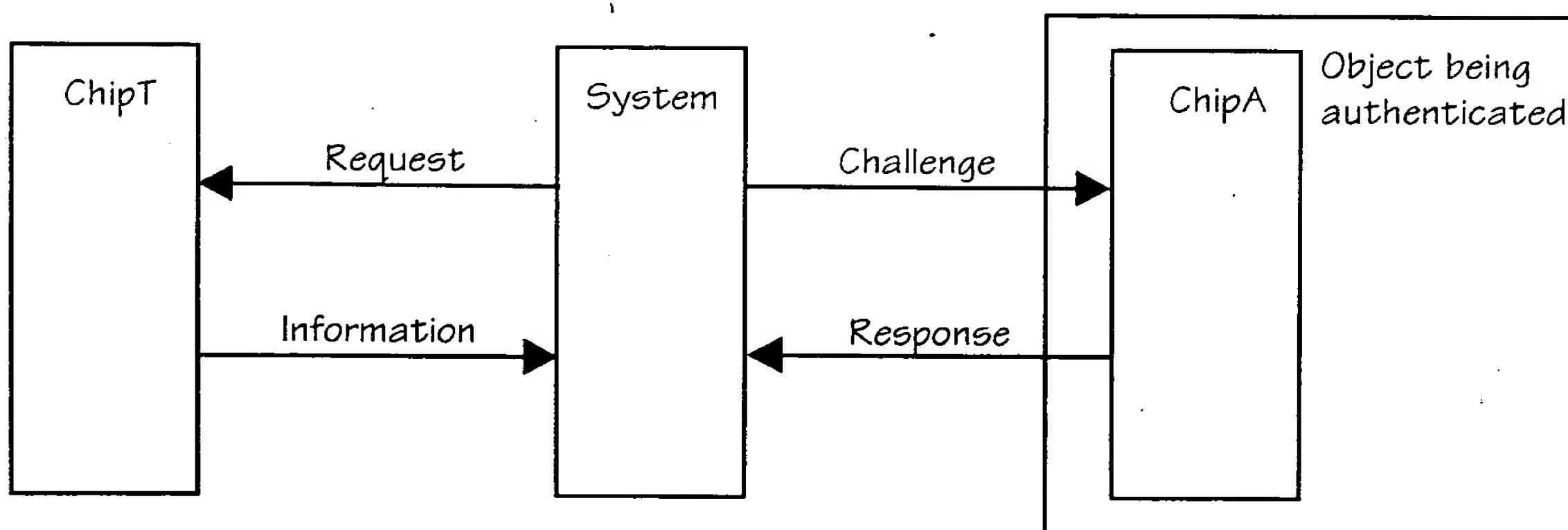


FIG. 168

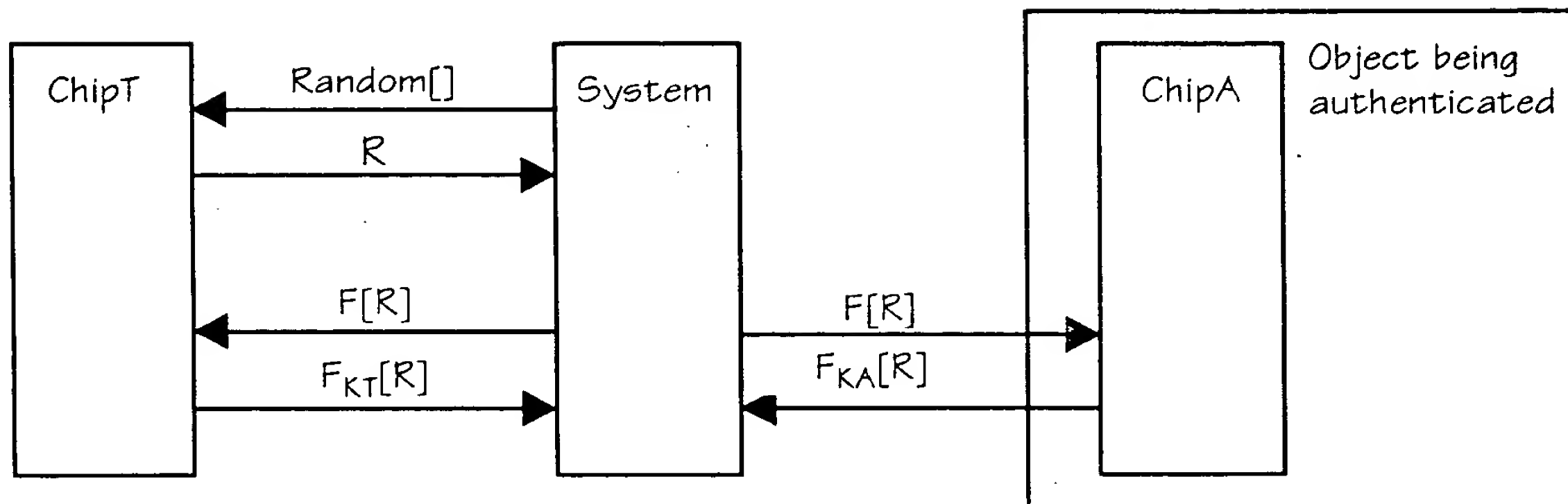


FIG. 169

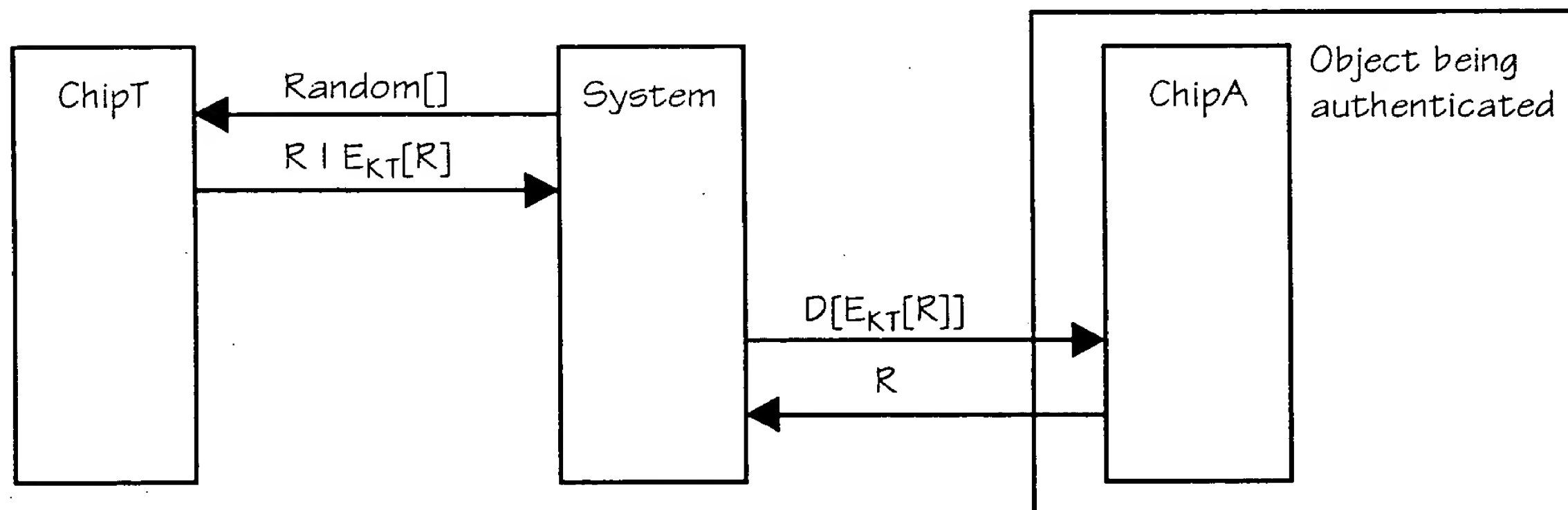


FIG. 170

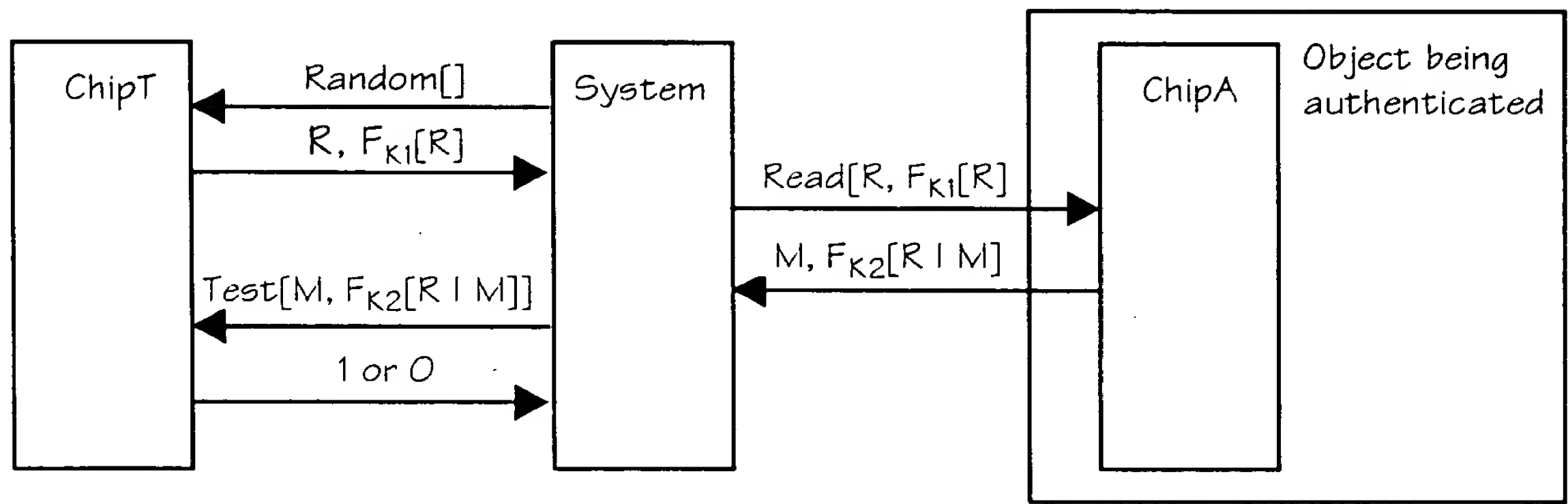


FIG. 171

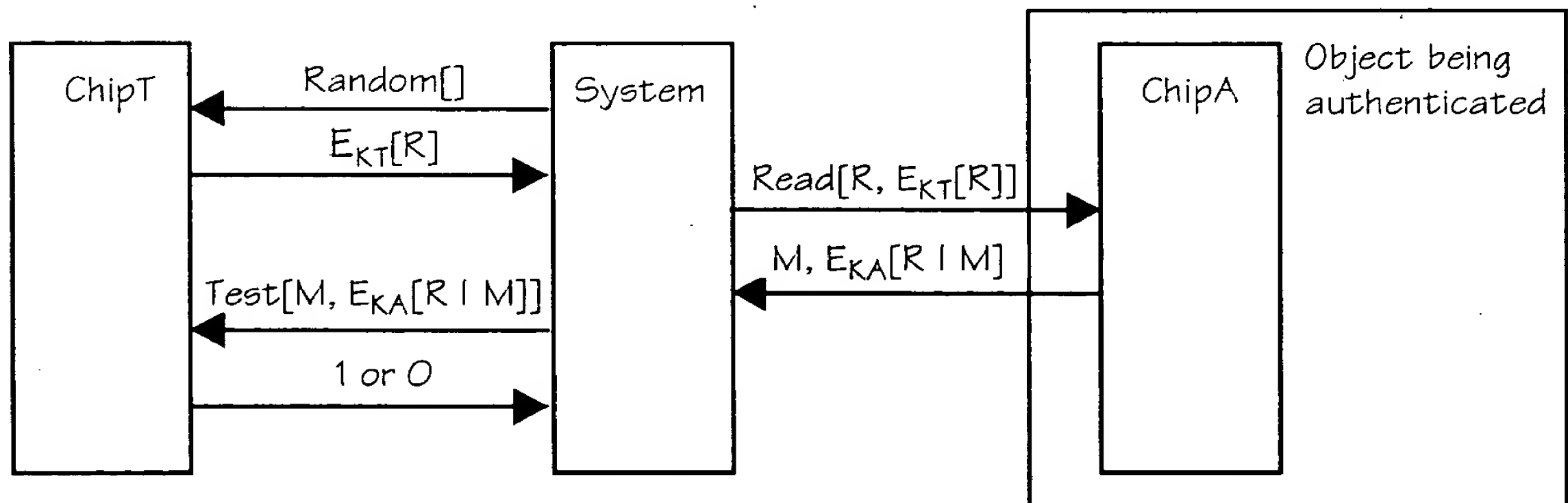


FIG. 172

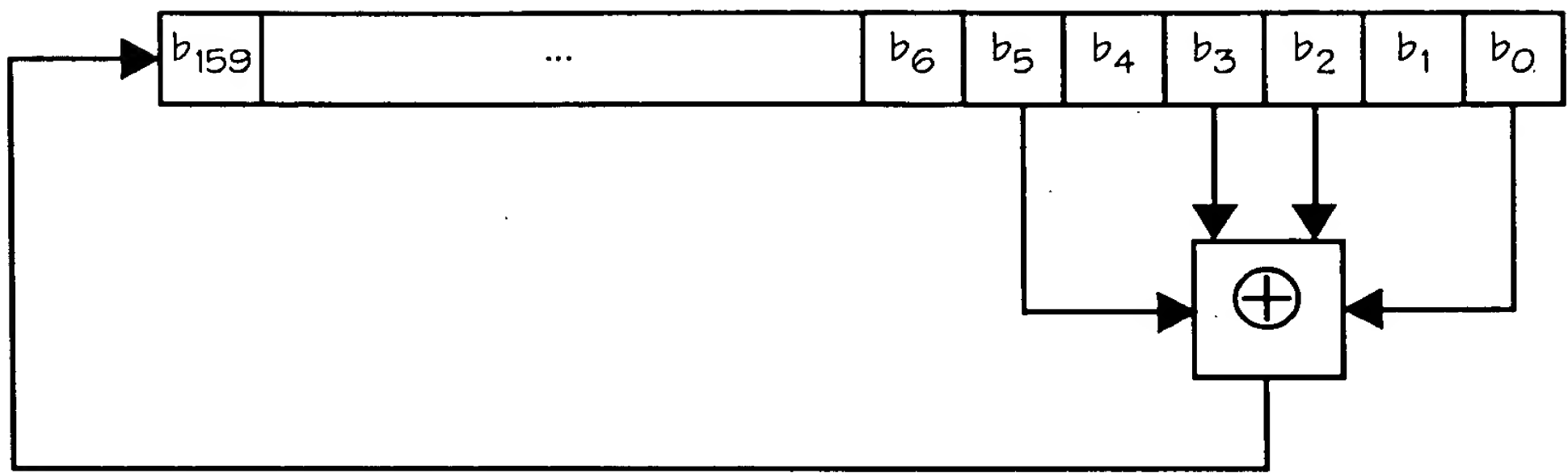


FIG. 173

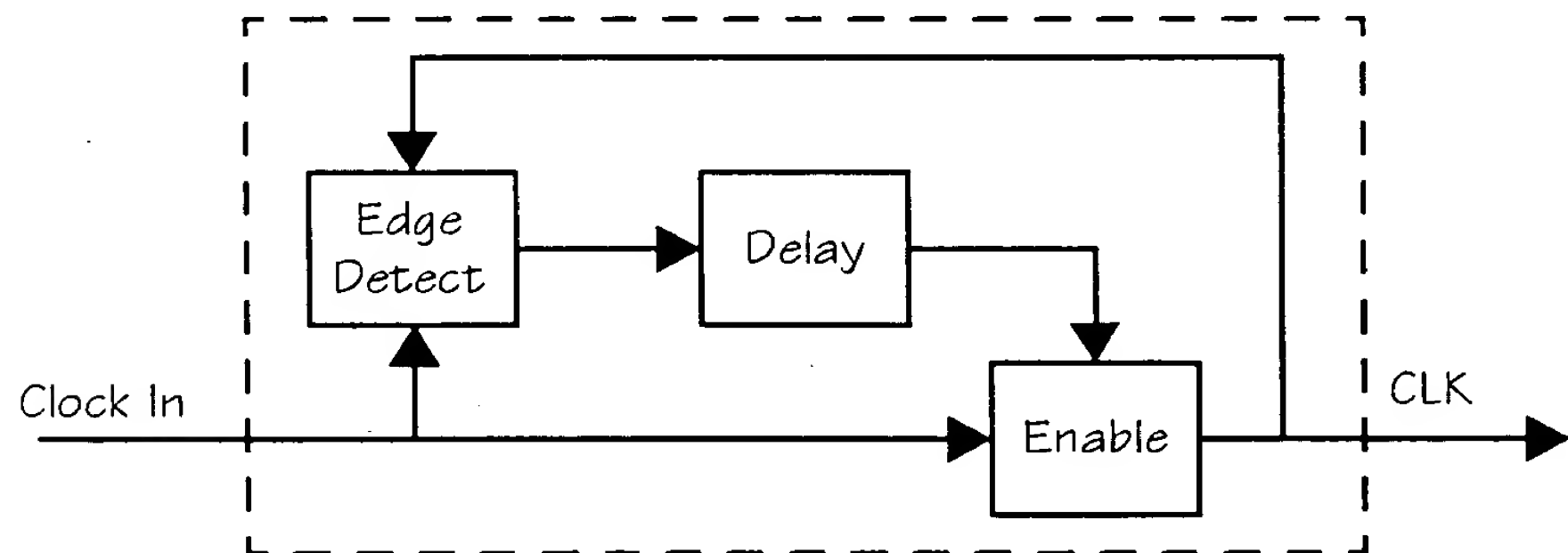


FIG. 174

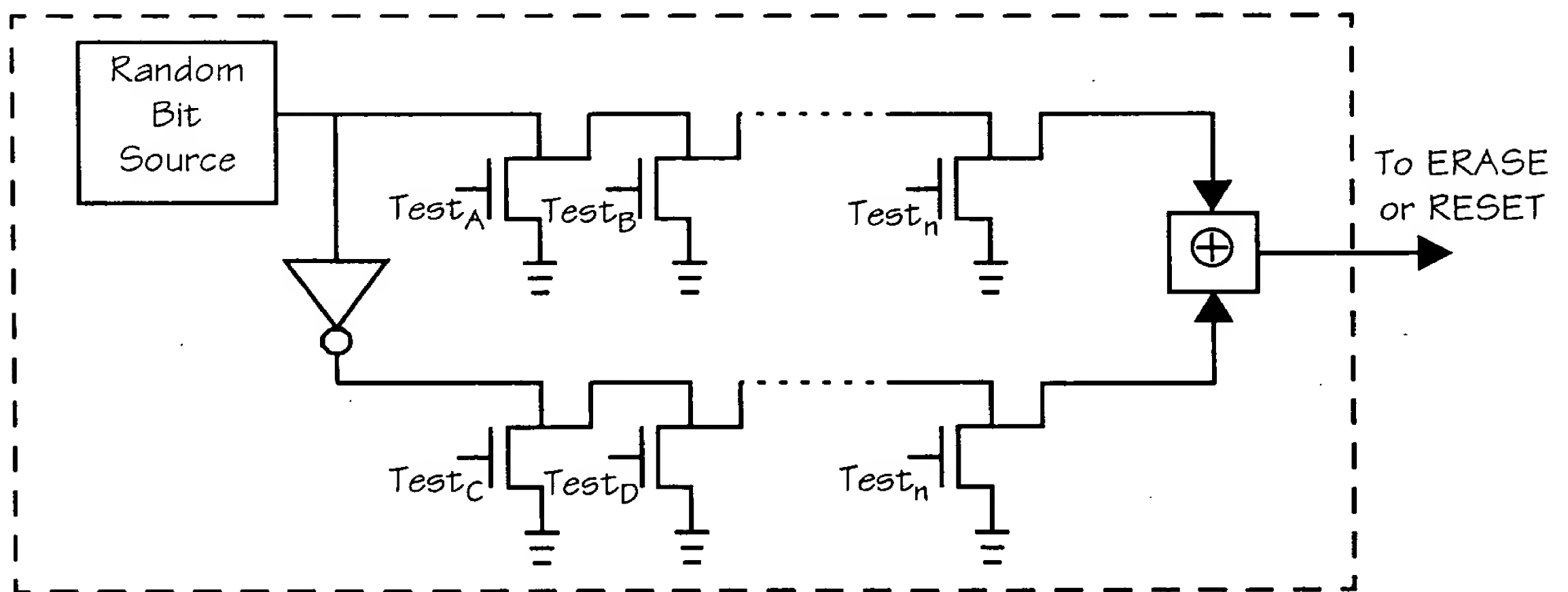


FIG. 175

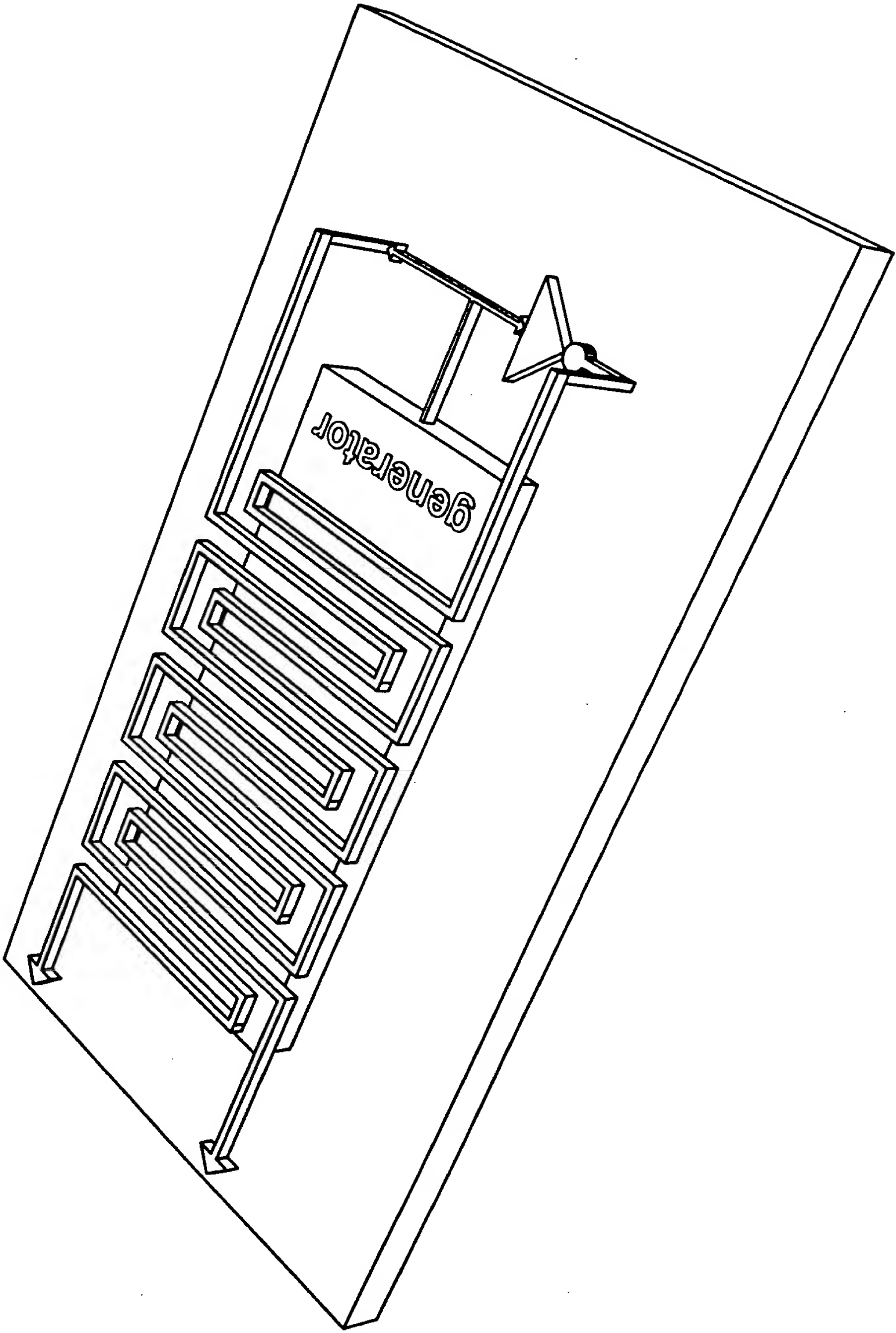


FIG. 178

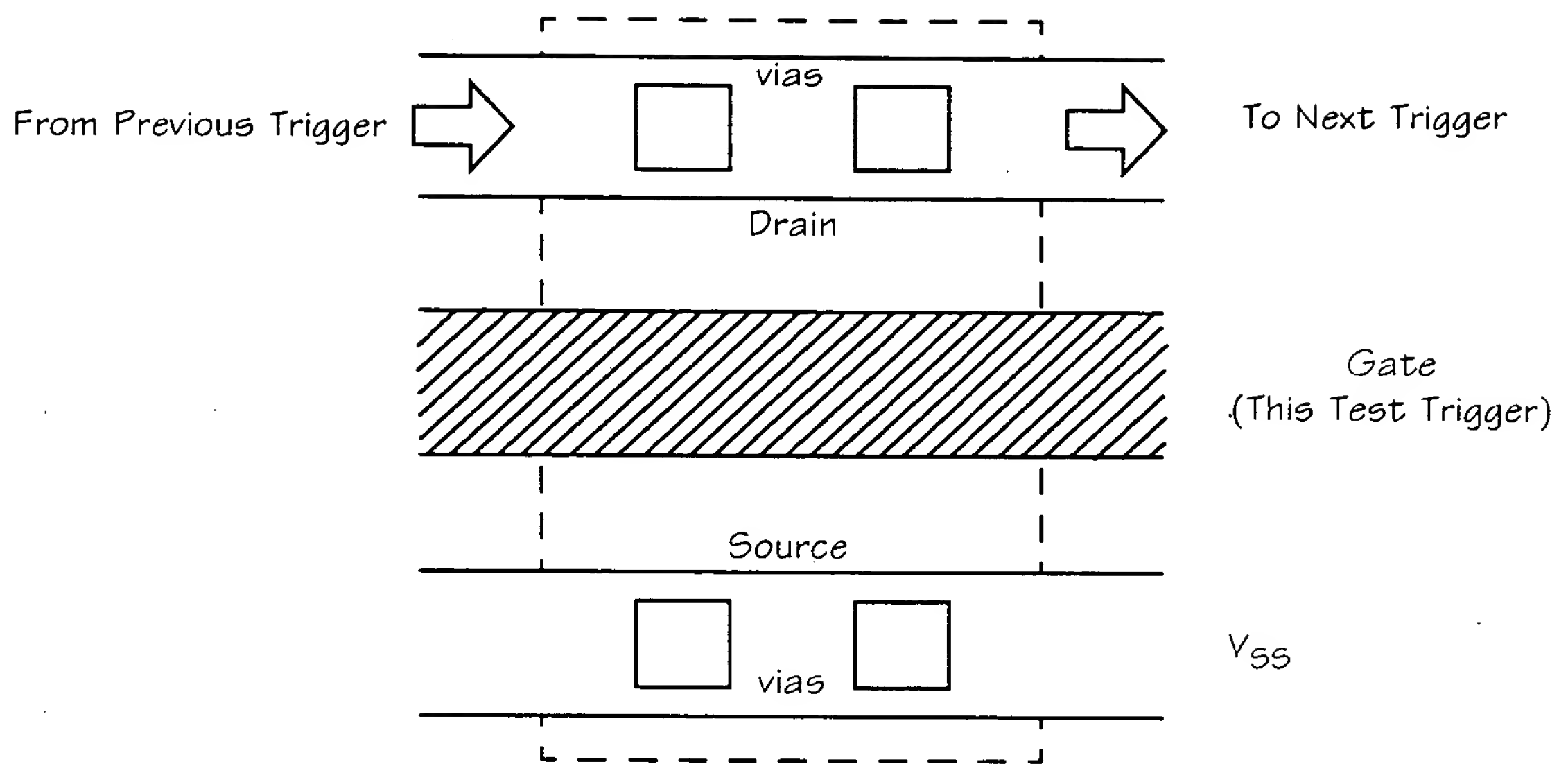


FIG. 176

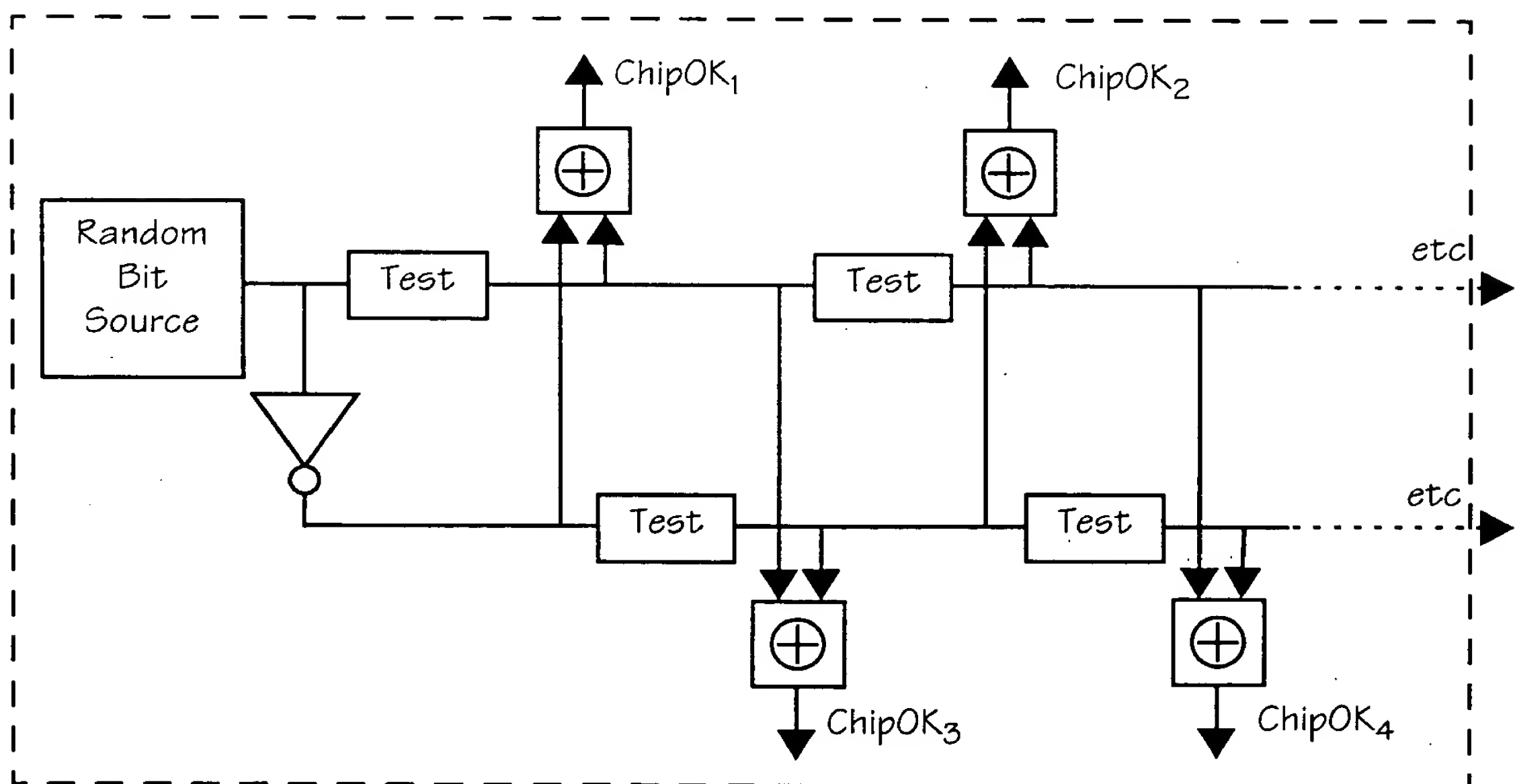


FIG. 177

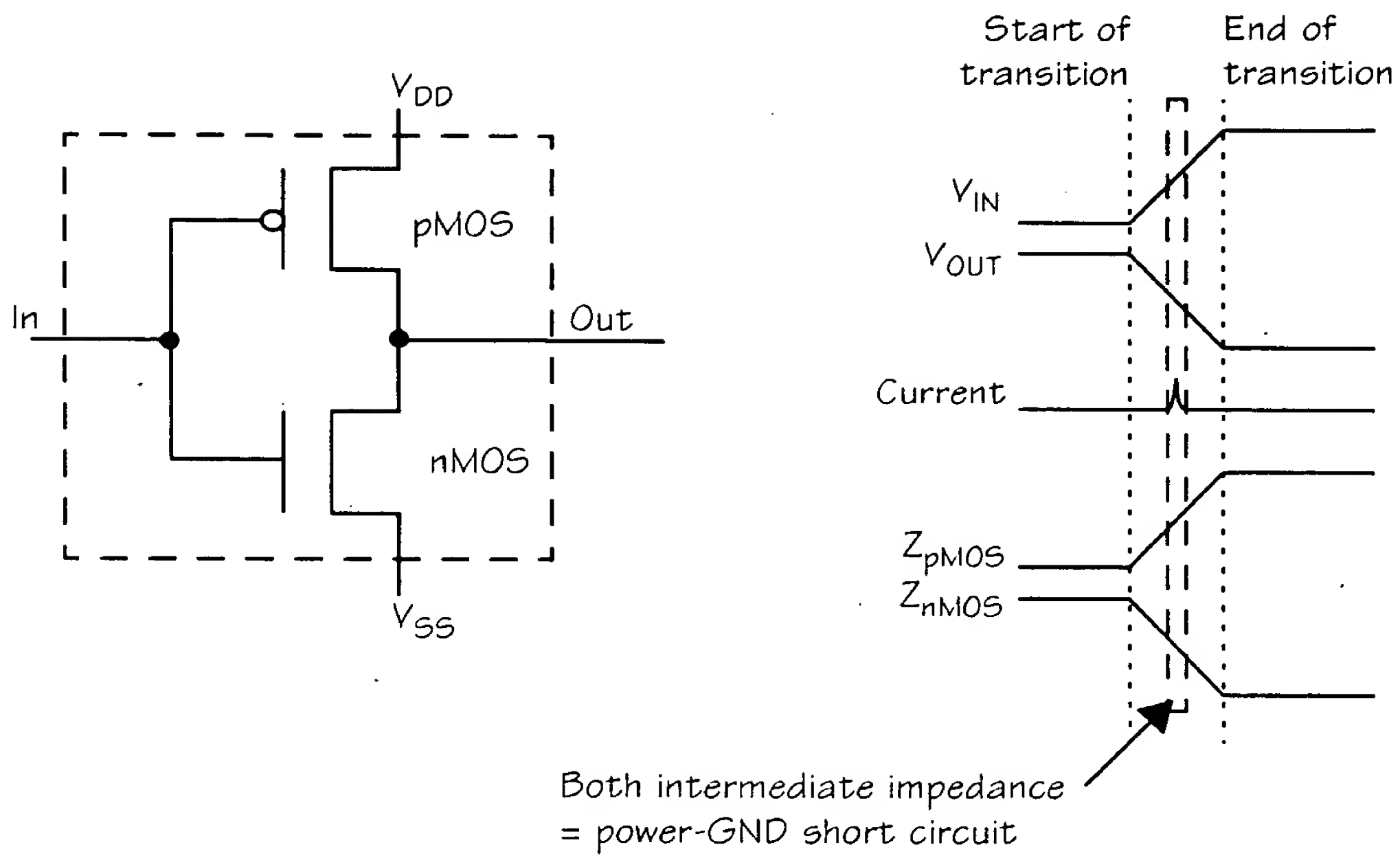


FIG. 179

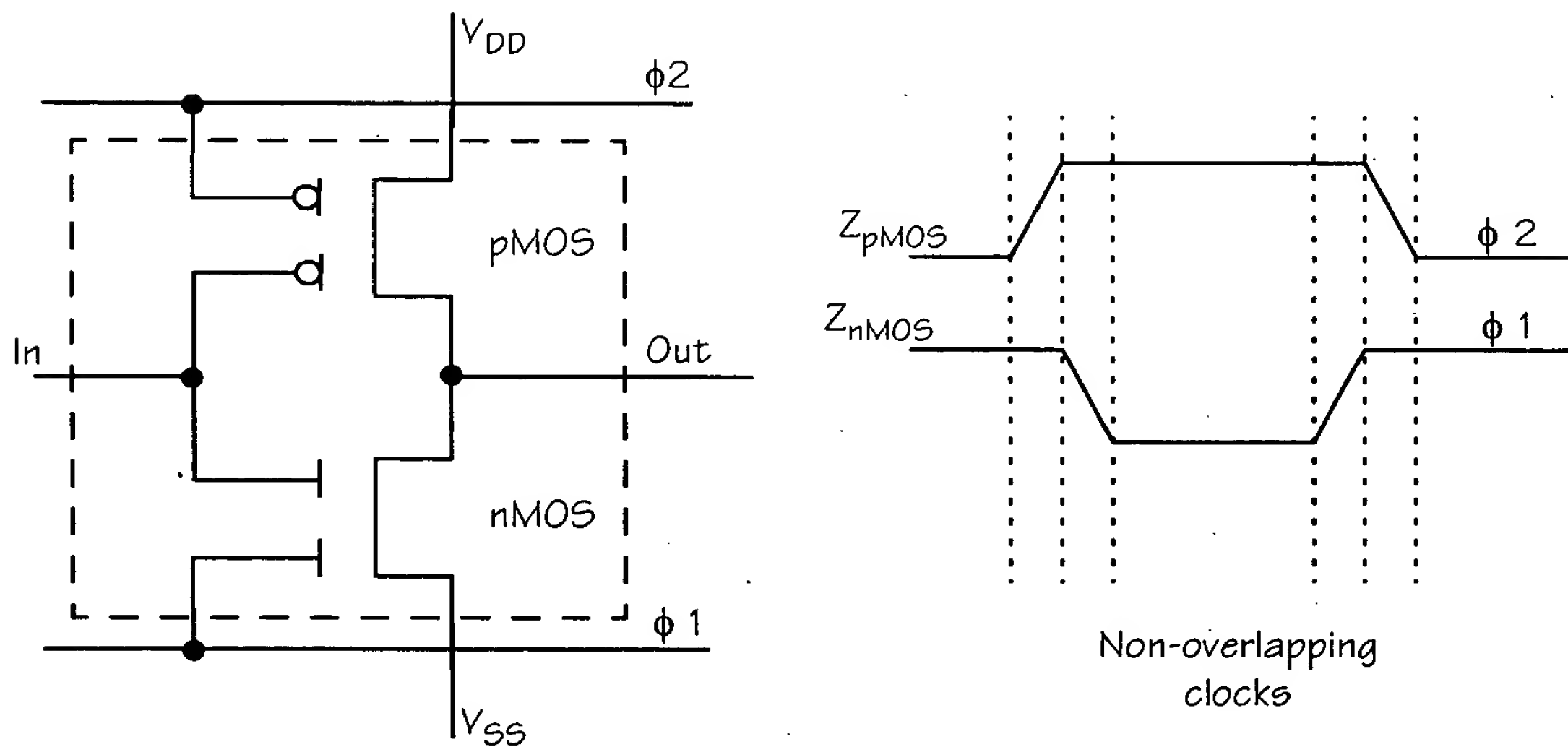


FIG. 180

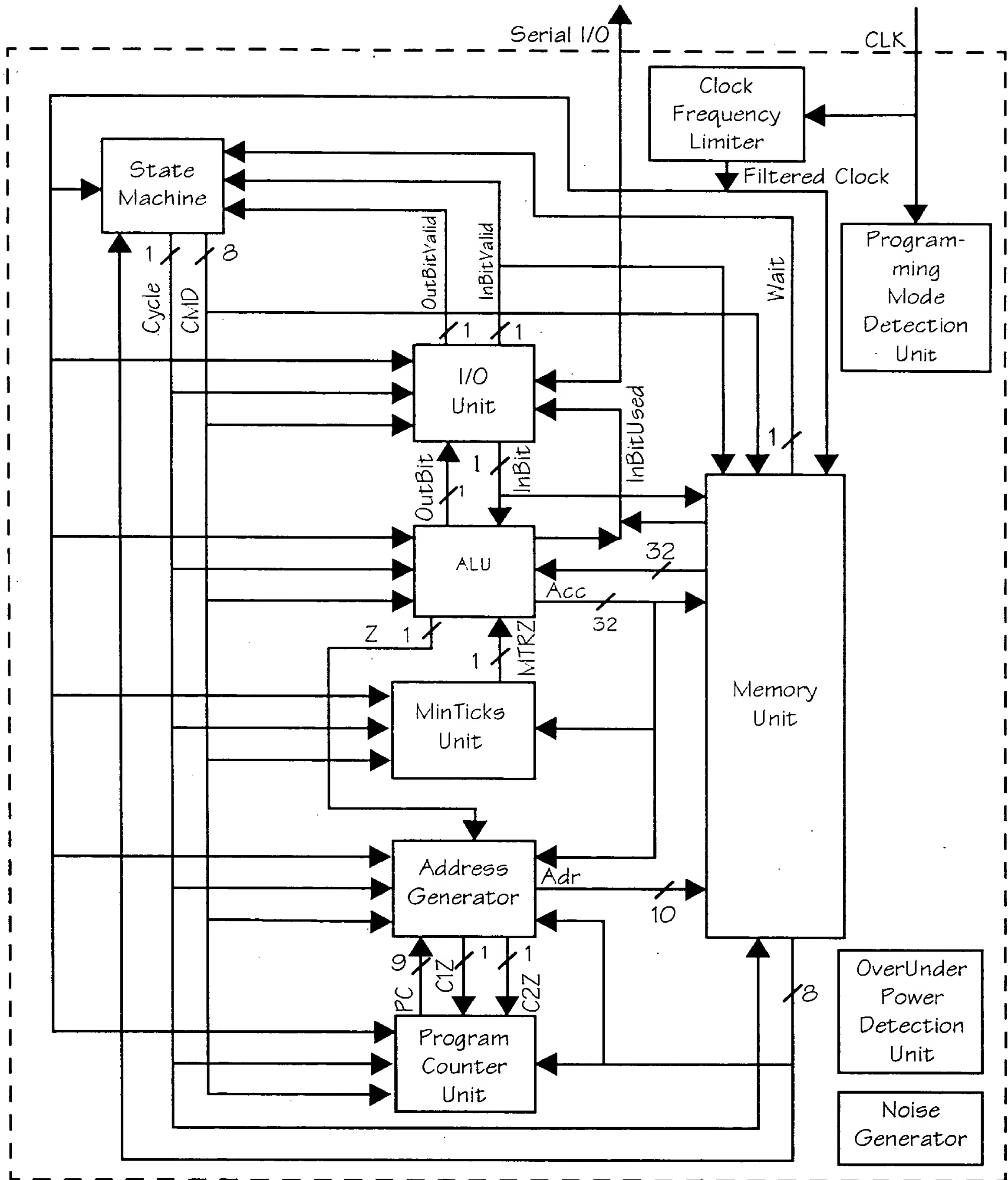


FIG. 181

FIG. 182

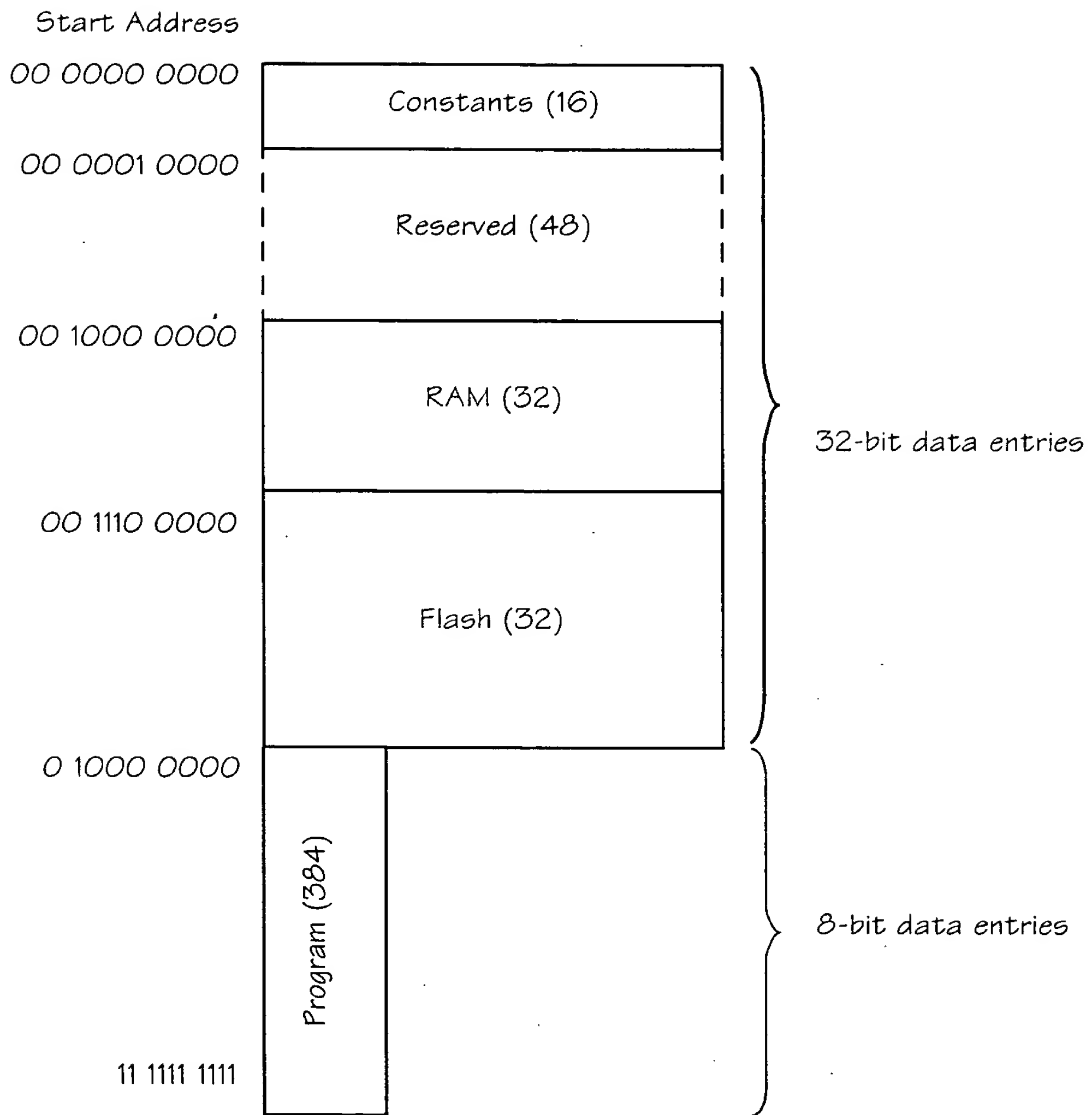


FIG. 182

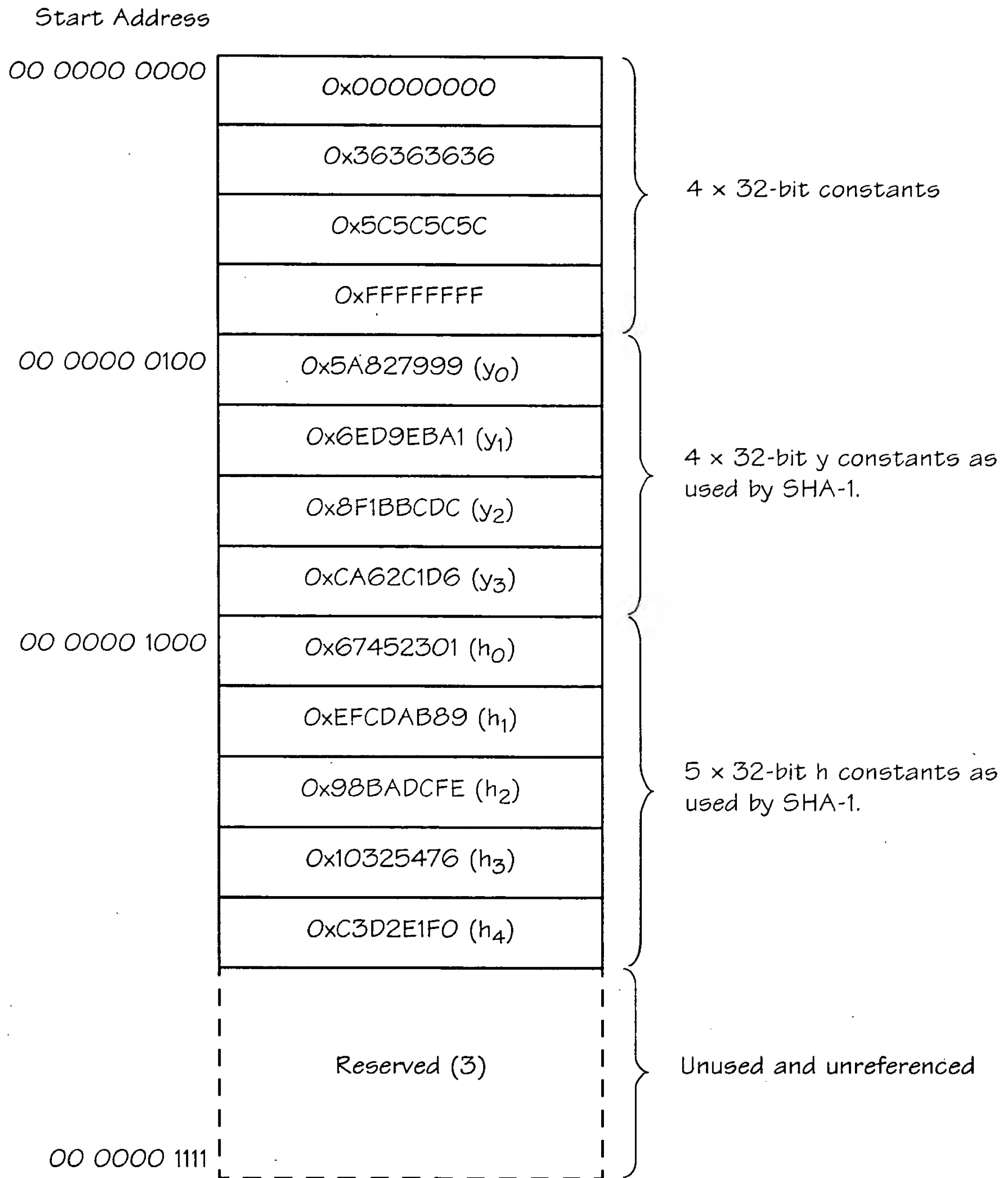


FIG. 183

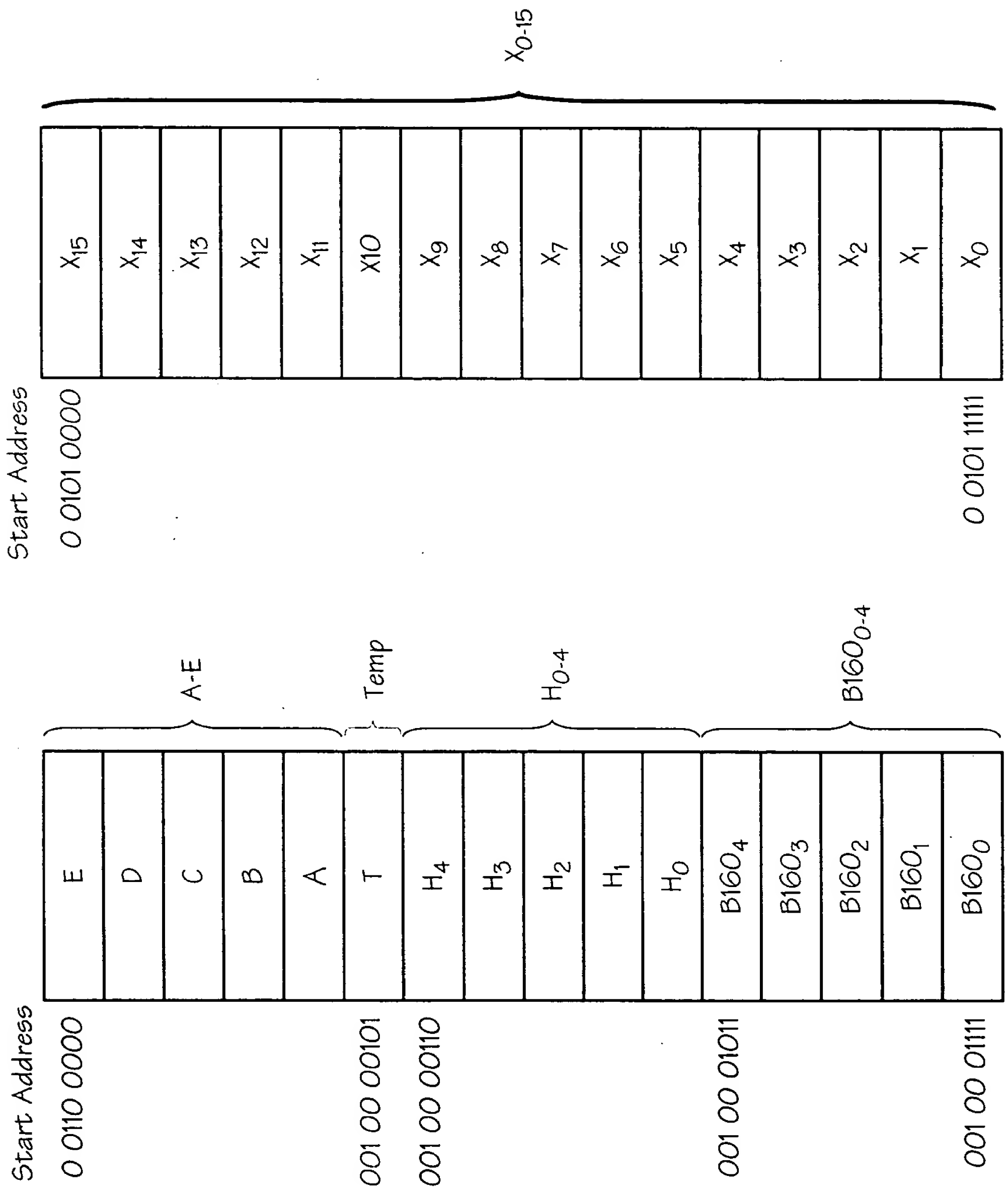


FIG. 184

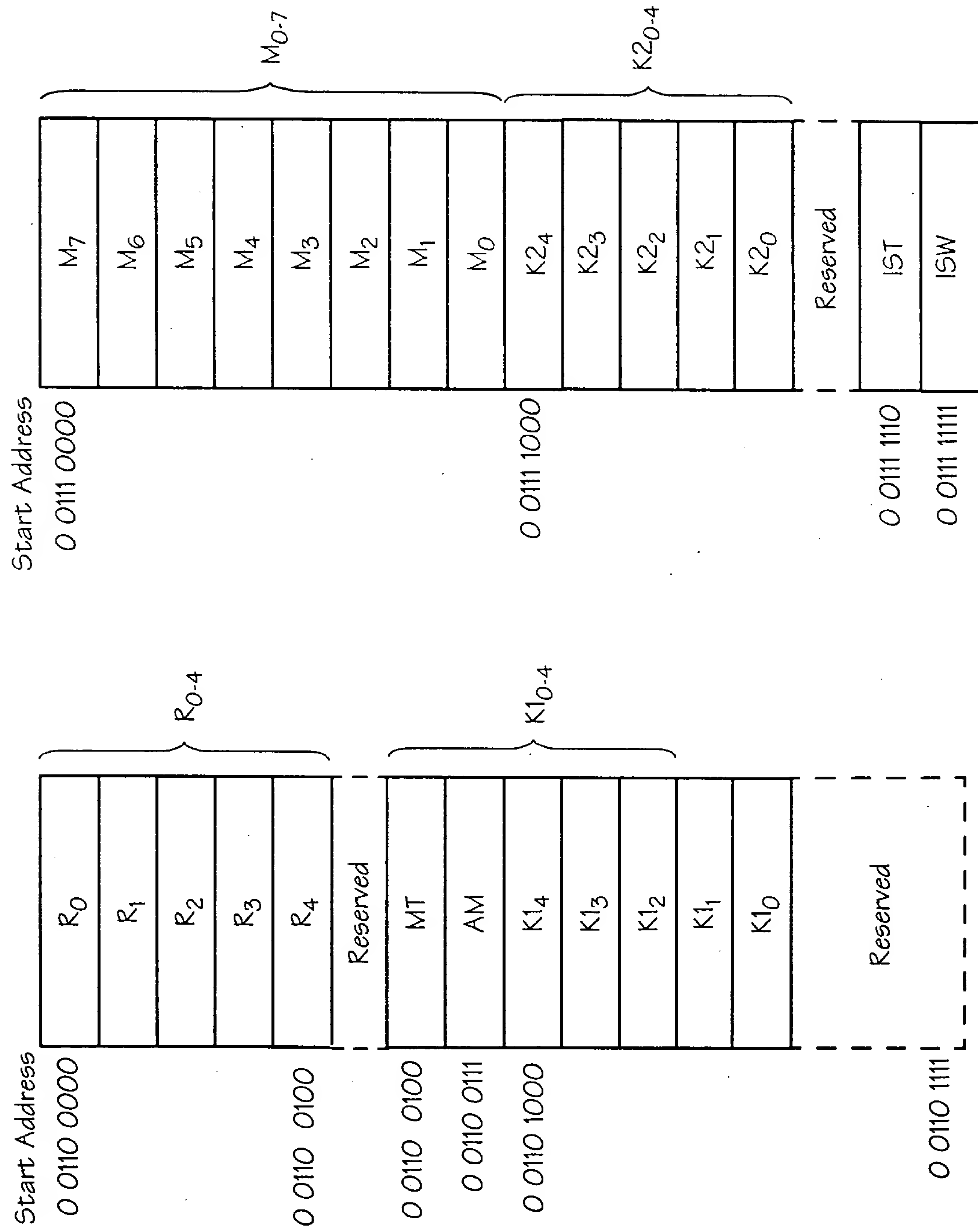


FIG. 185

Start Address

Adr Table 1 (32)

DBR Table (8)

11 1111 1111

FIG. 186

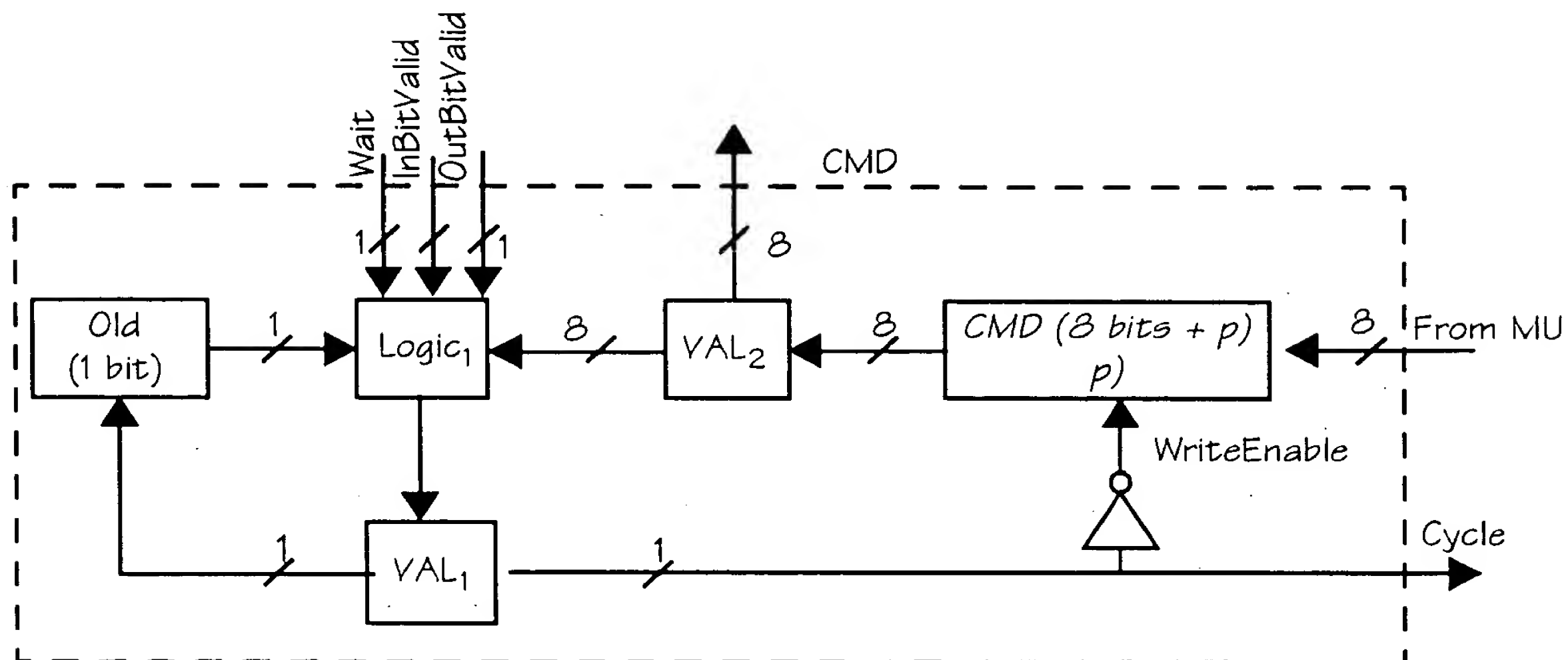


FIG. 187

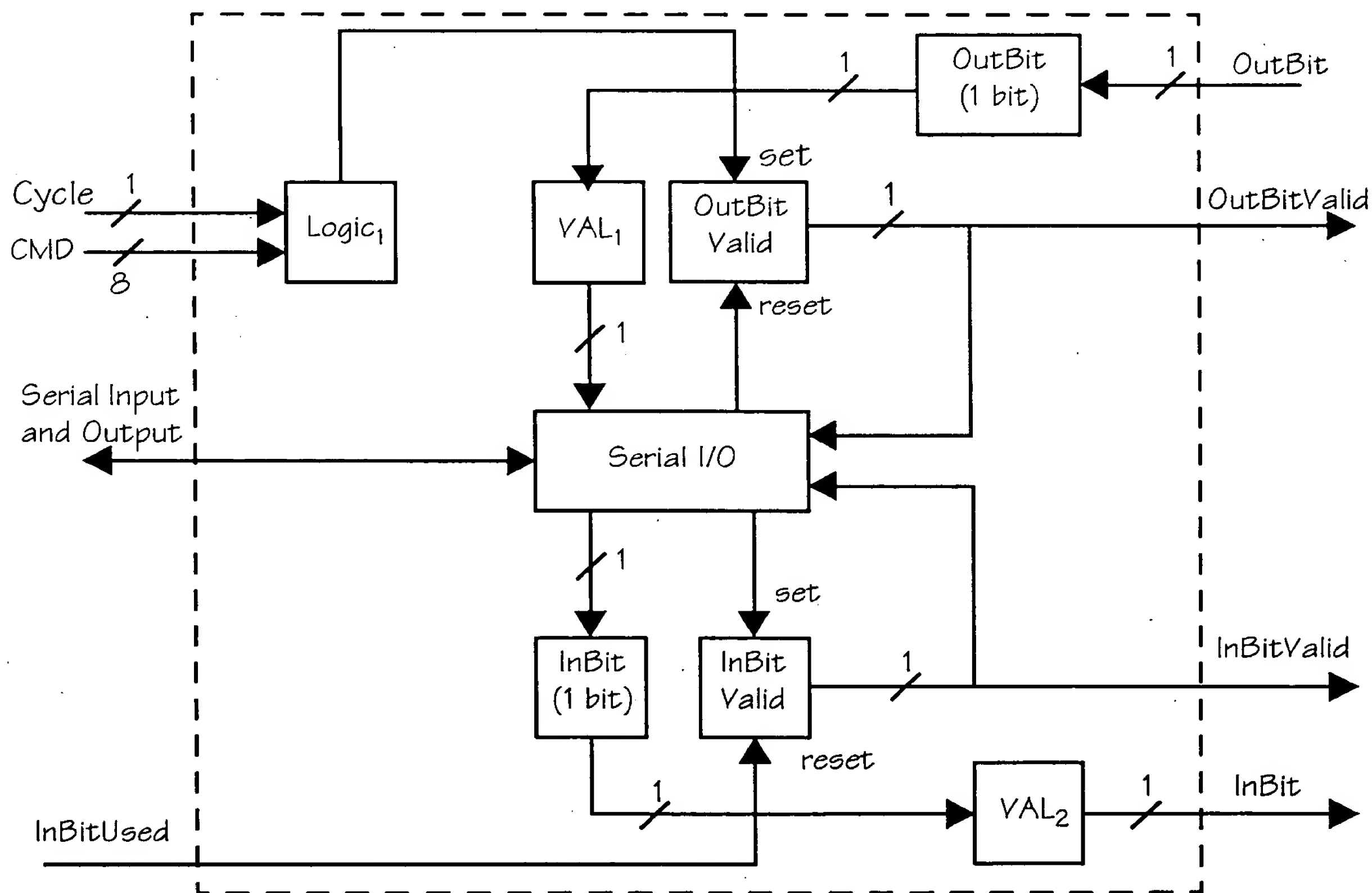


FIG. 188

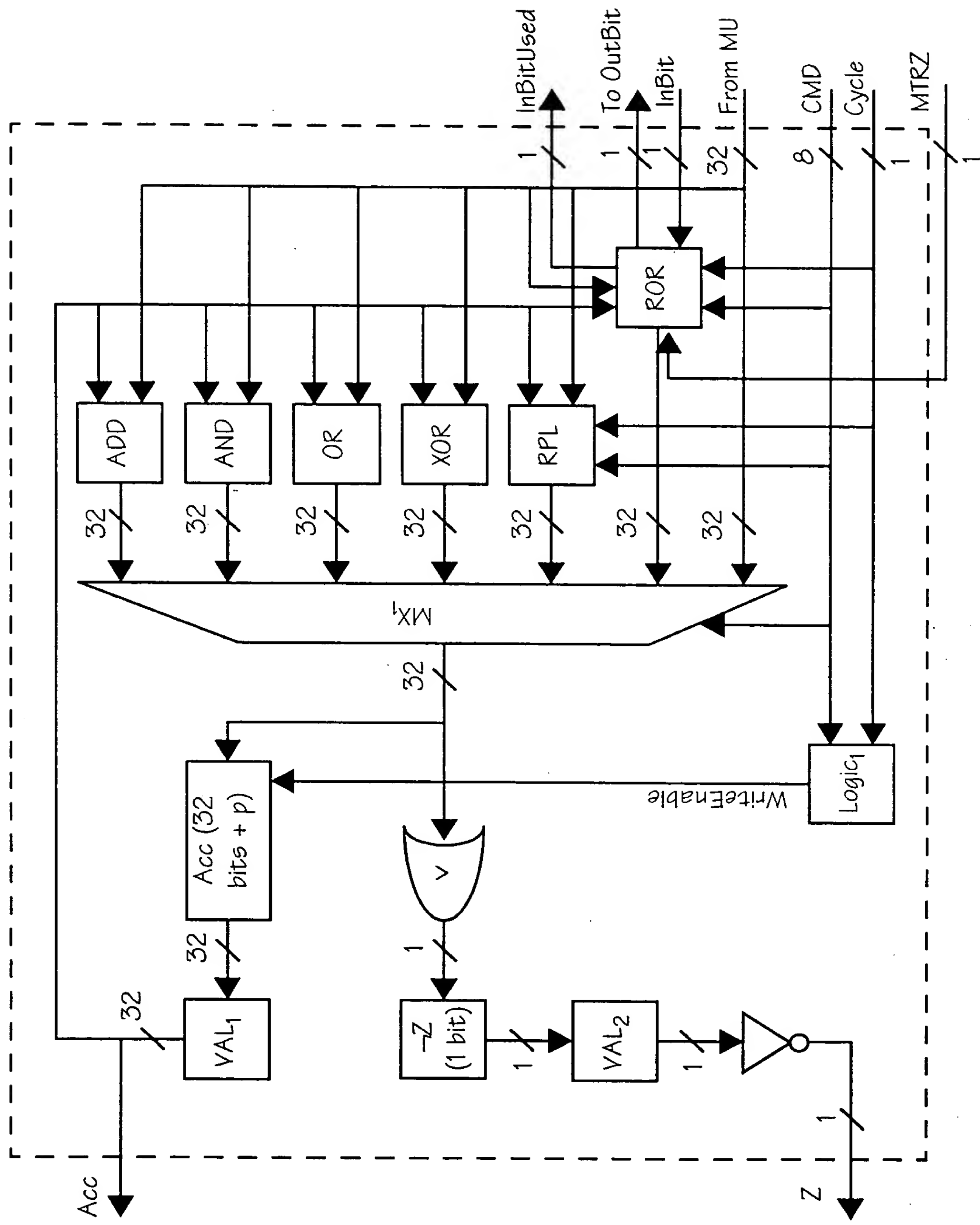


FIG. 189

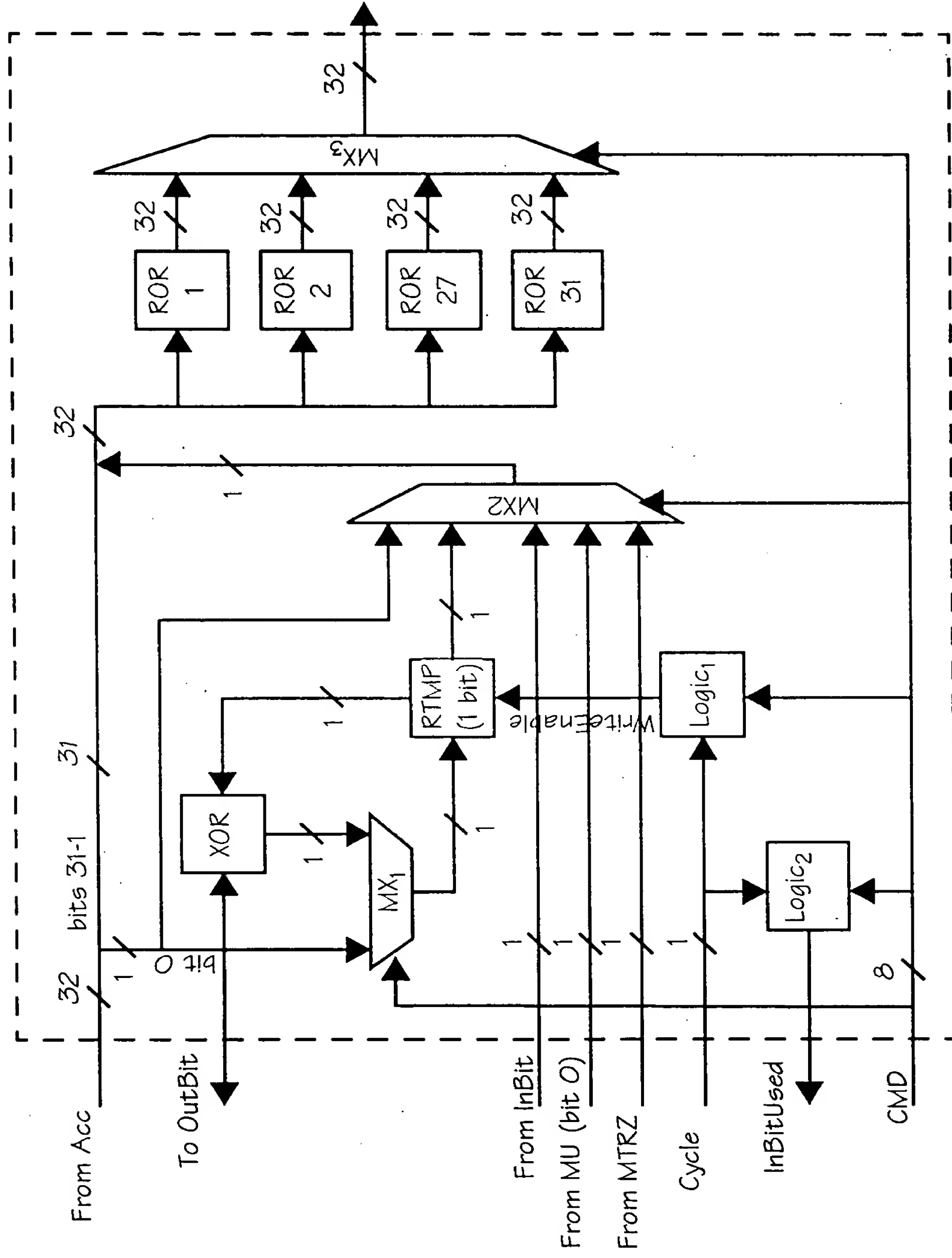


FIG. 190

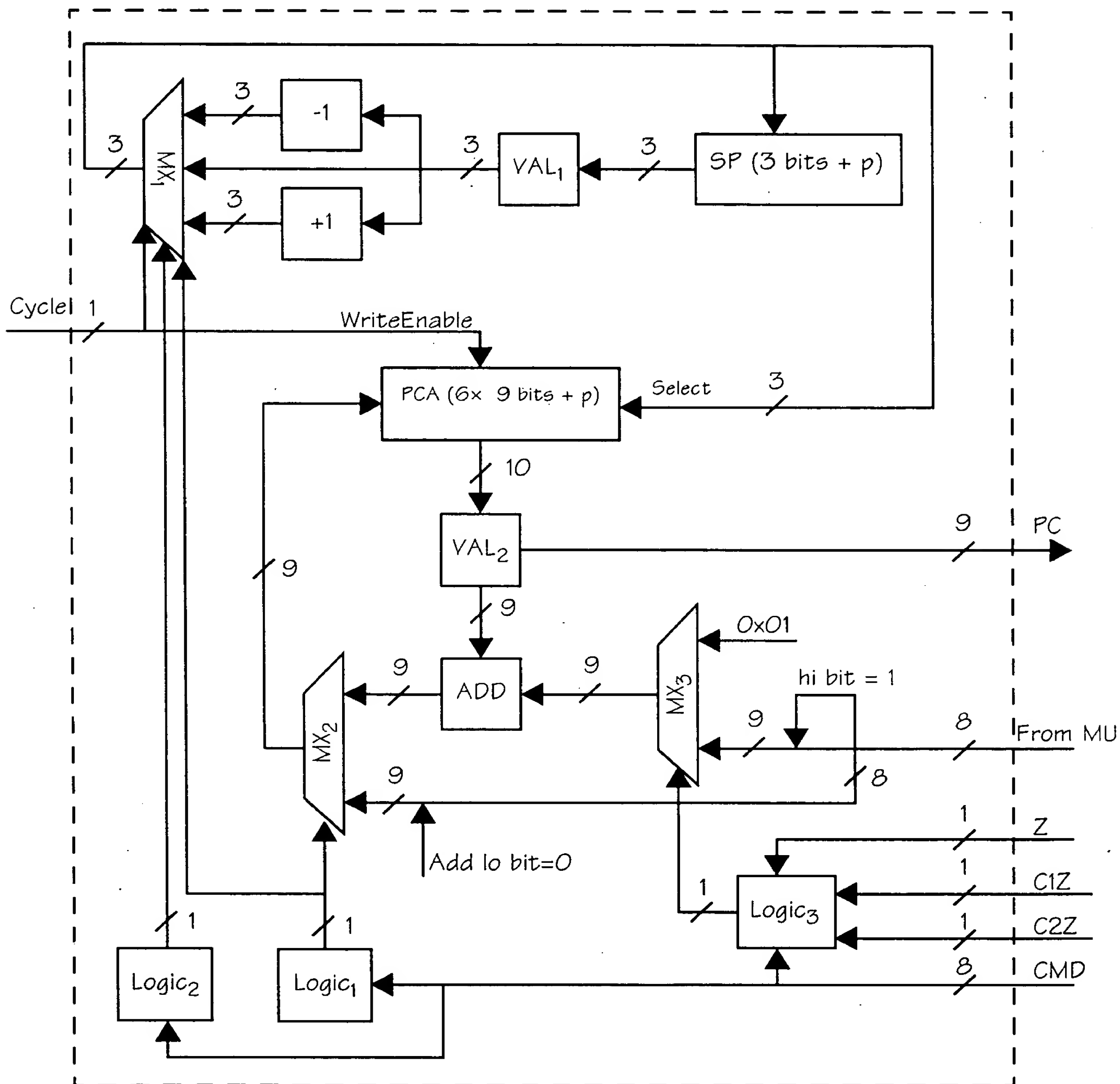


FIG. 192

FIG. 193

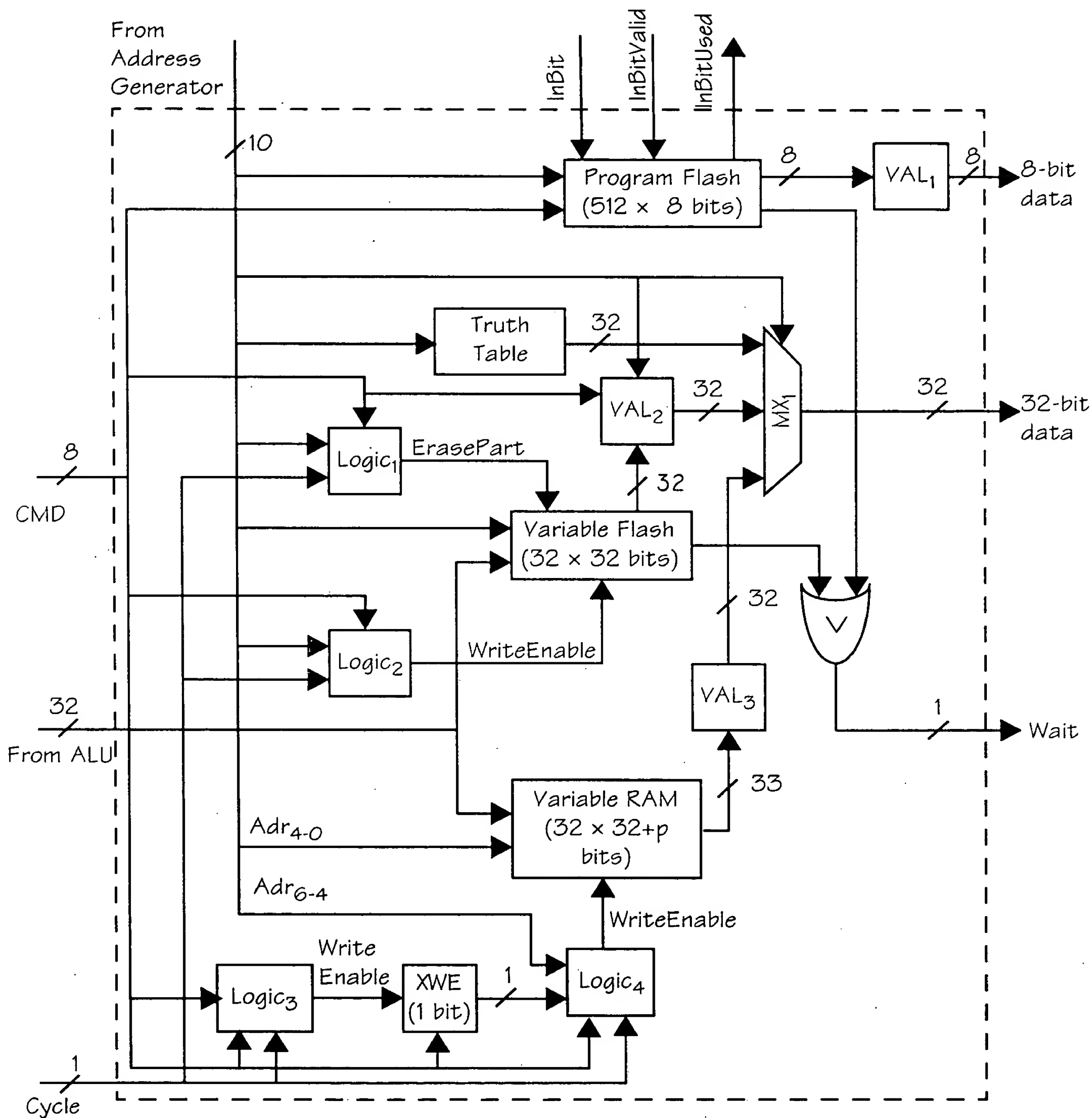


FIG. 193

TOP SECRET

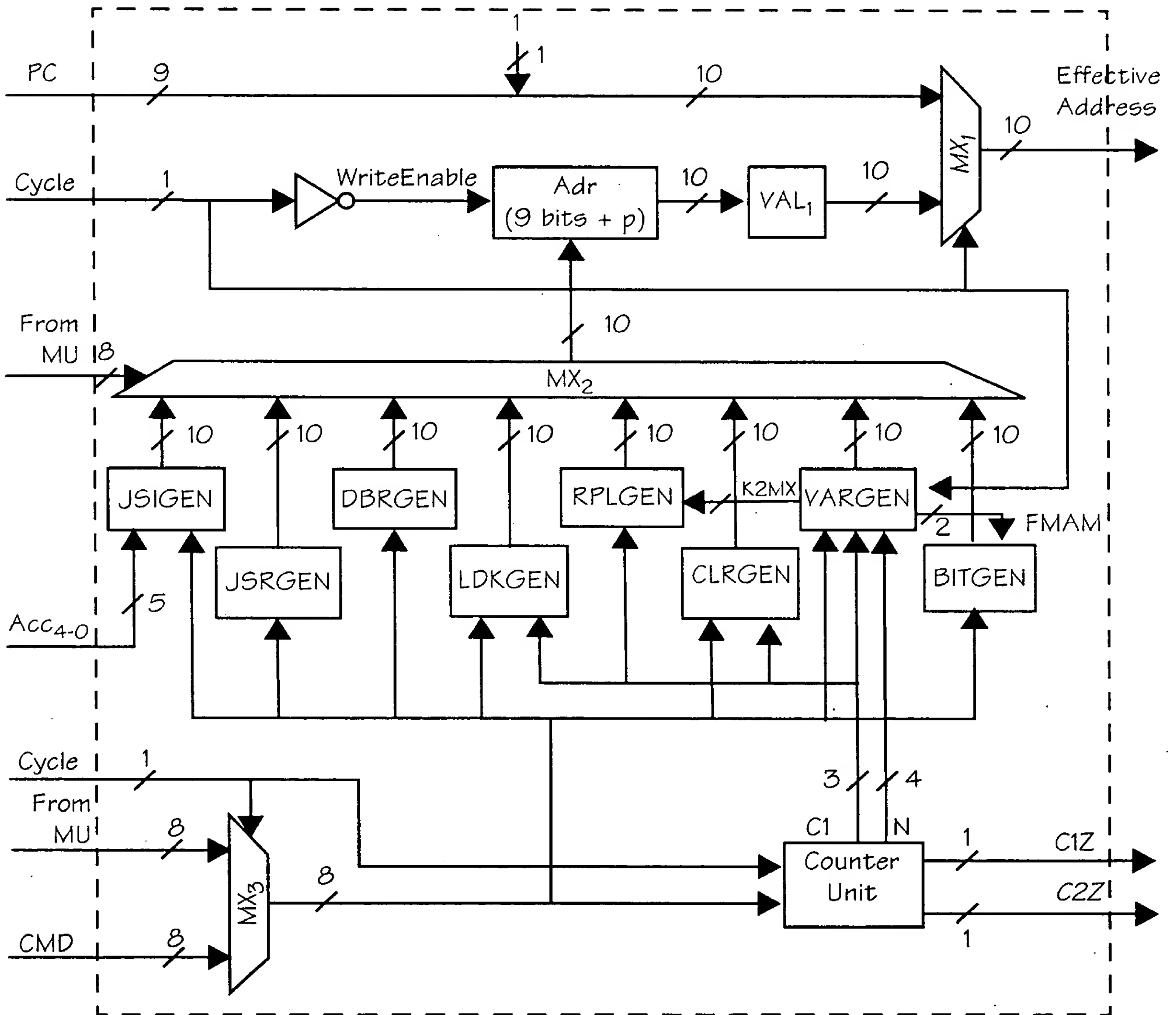


FIG. 194

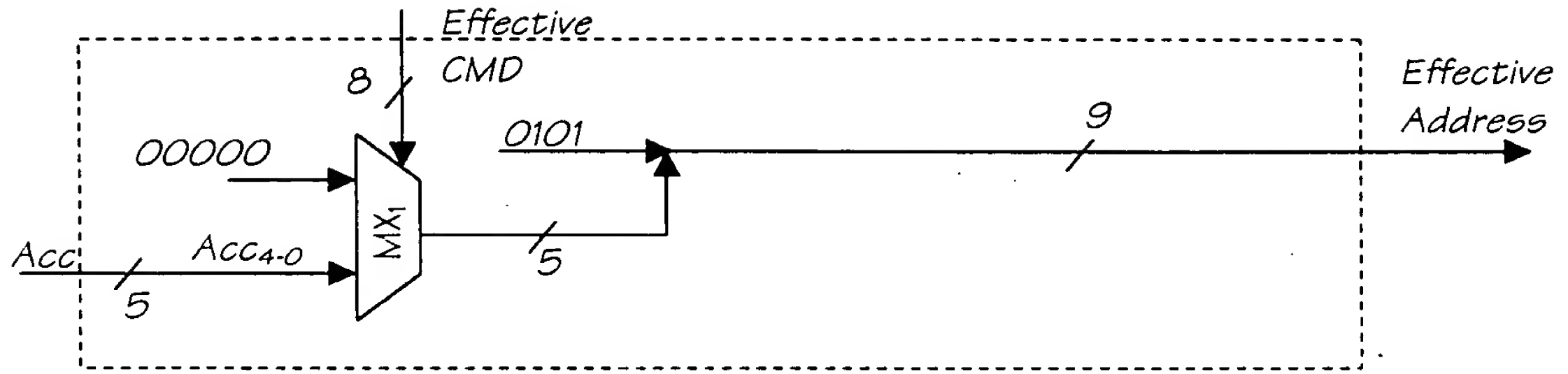


FIG. 195

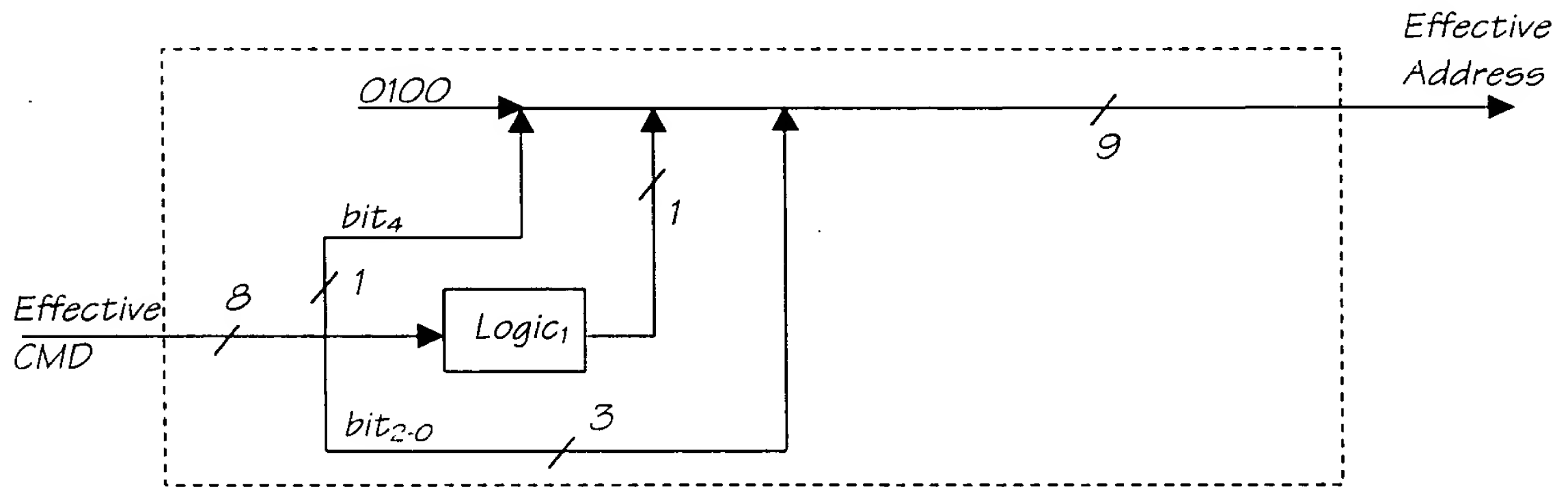


FIG. 196

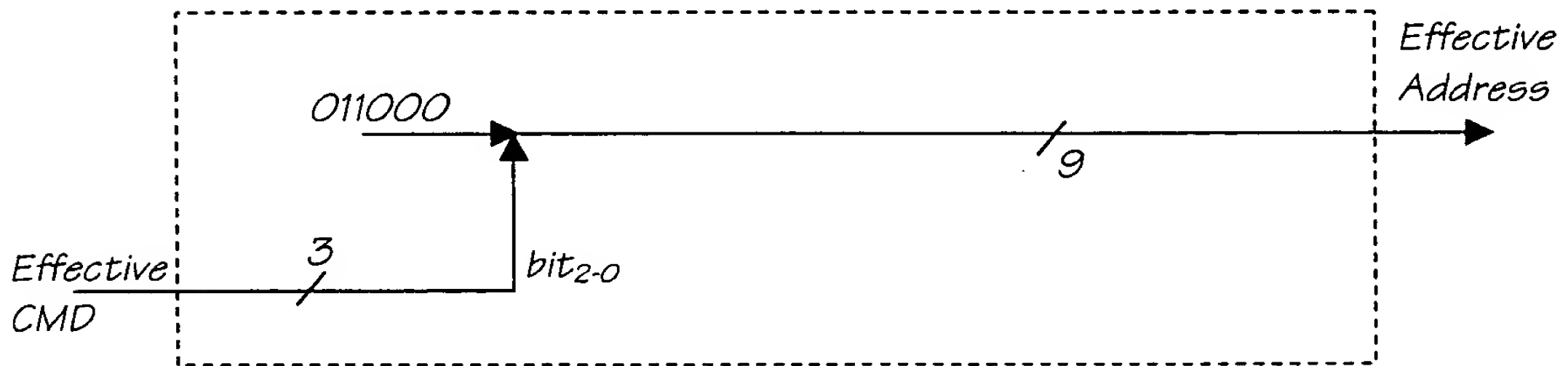


FIG. 197

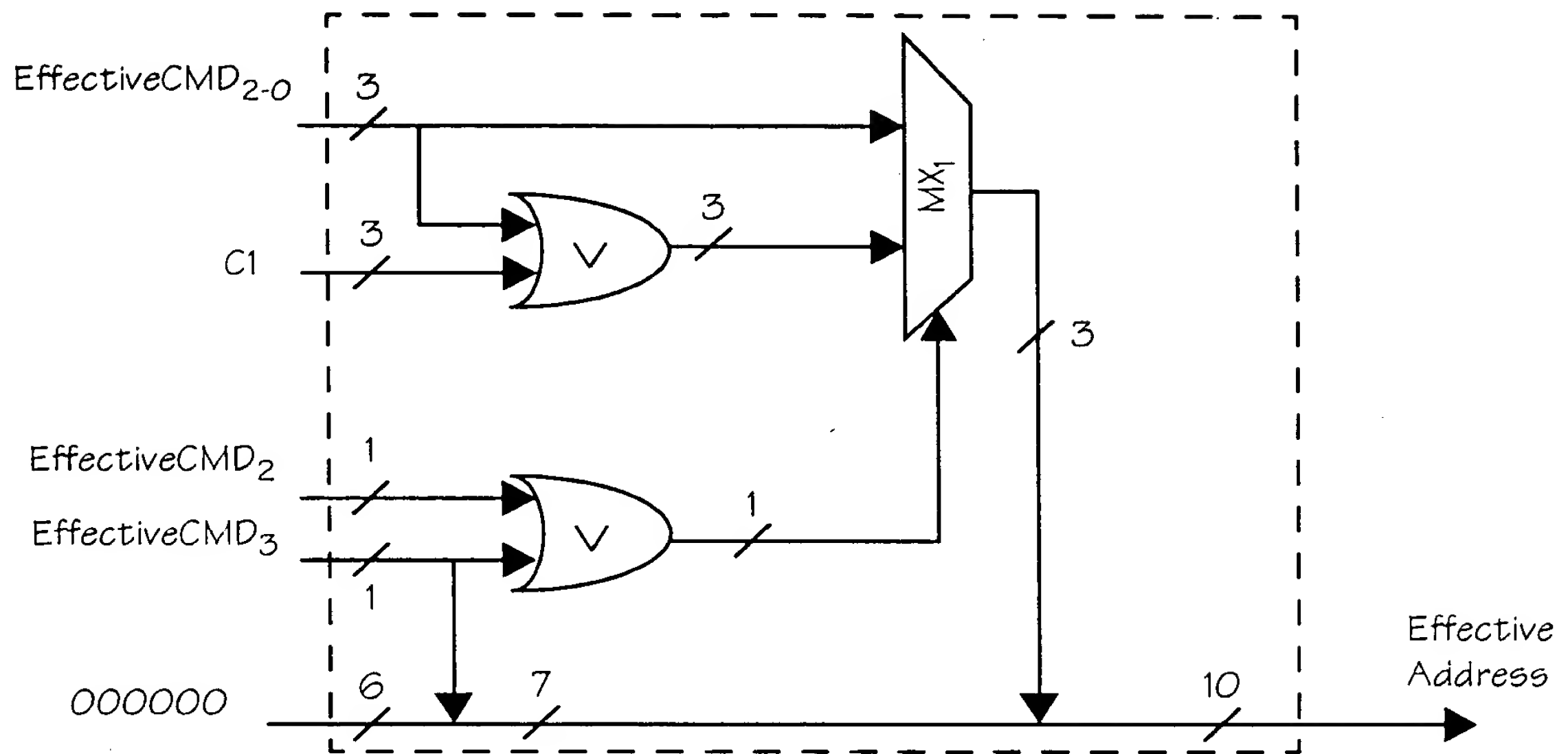


FIG. 198

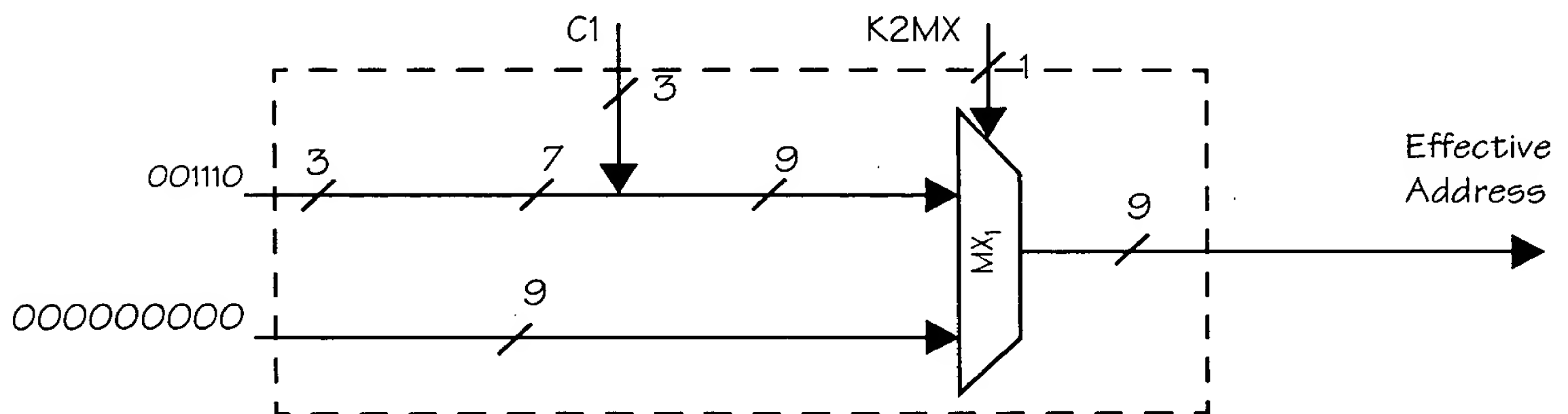


FIG. 199

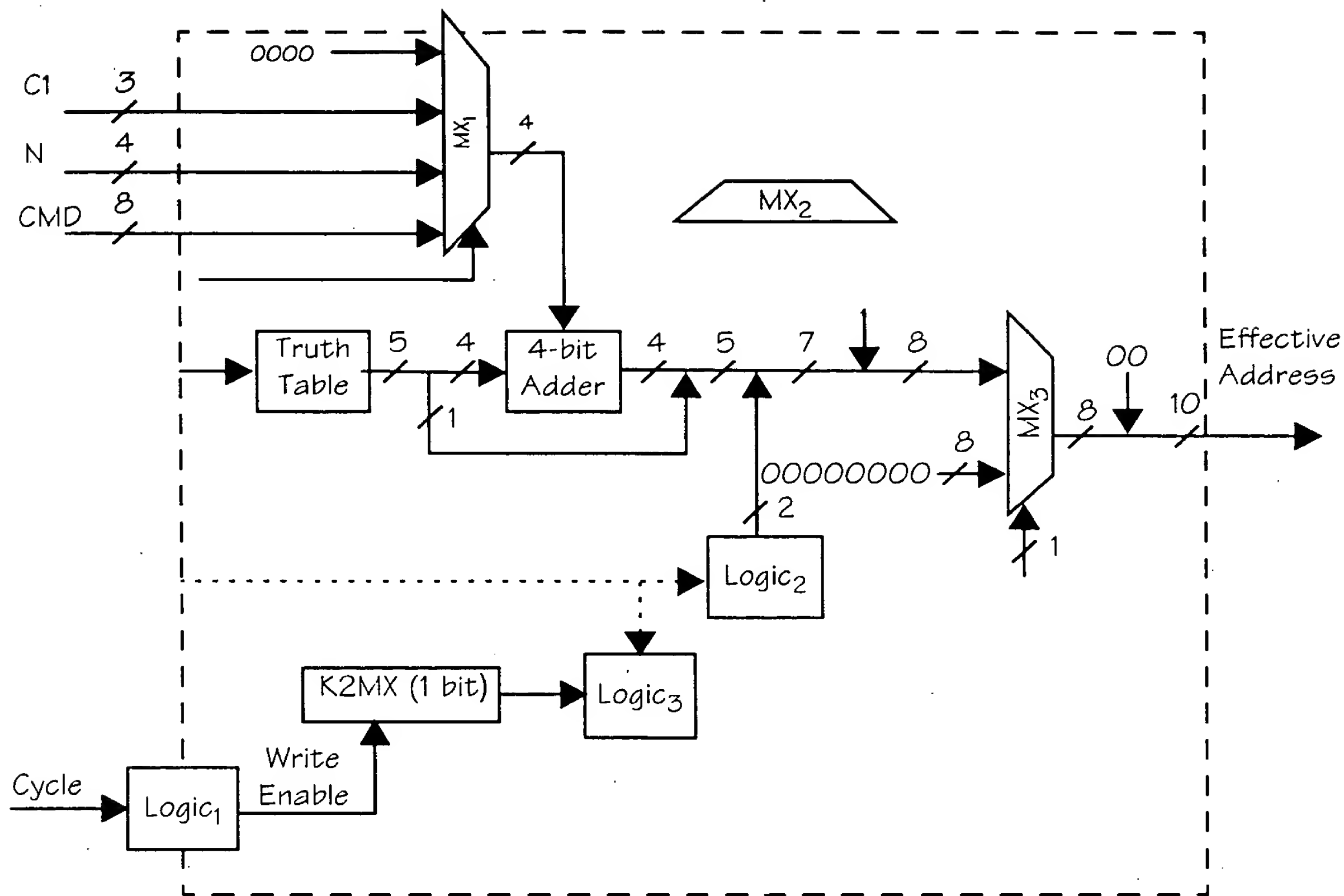


FIG. 200

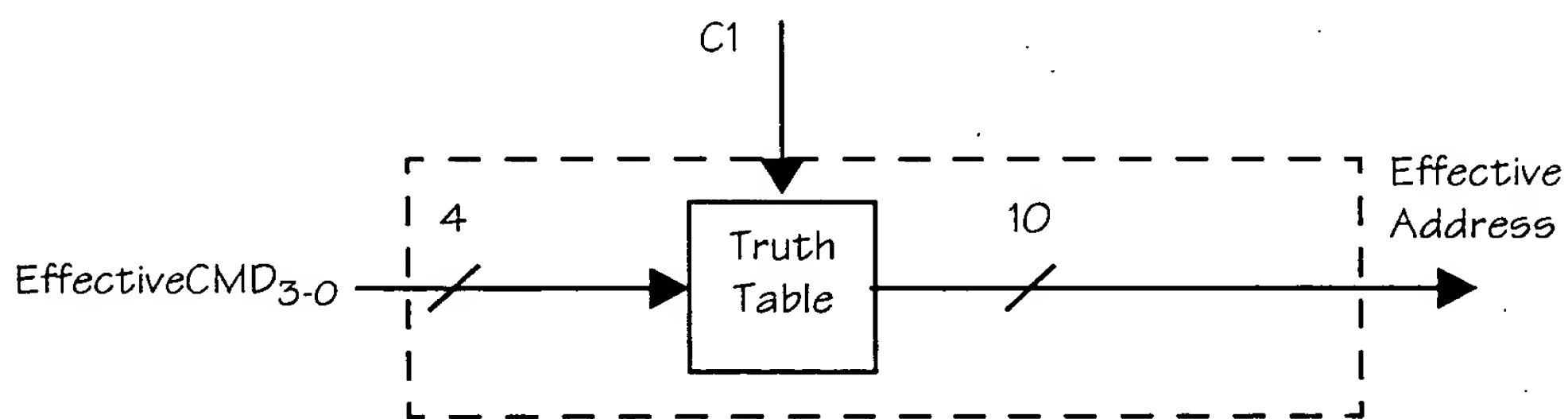


FIG. 201

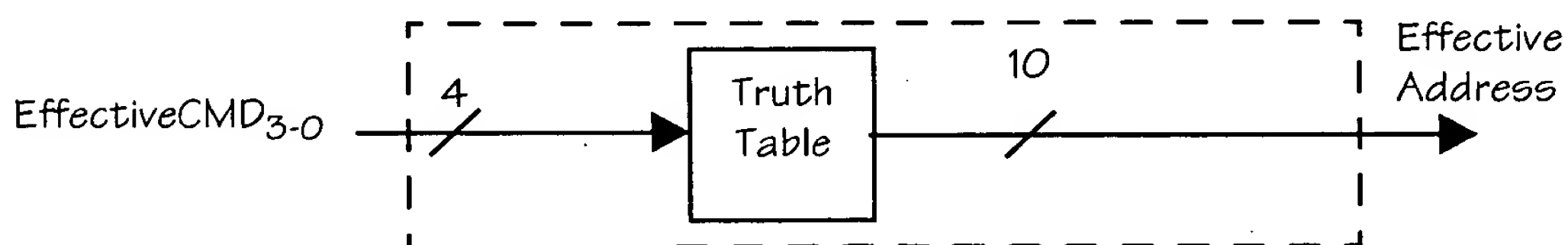


FIG. 202

The diagram illustrates the control logic for a 16-bit parallel adder. It features several key components and signal paths:

- Inputs:** *CMD* (8 bits) and *Cycle* (1 bit).
- Logic Blocks:** *Logic1*, *Logic2*, *Logic3*, *Logic4*, *Logic5*, *Logic6*, and *Logic7* are distributed throughout the circuit to manage data flow and control signals.
- Multiplexers:** *MX1*, *MX2*, *MX3* (four instances), and *MX4* are used to select between different data paths based on control signals.
- Registers and Counters:**
 - C1* (3 bits + *p*) and *C2* (5 bits + *p*) are counters or registers that track the current bit position.
 - N1* through *N4* (4 bits + *p*) are registers that store intermediate results or control data.
 - VAL1*, *VAL2*, and *VAL3* are value registers that hold specific control values.
- Control Signals and Operations:**
 - Write Enable* signals are used to control data flow into and out of registers.
 - Truth Table* and *Logic3* provide specific control logic based on the *CMD* input.
 - Logic4* through *Logic7* manage the *Write Enable* signals for registers *N1* through *N4*.
 - MX3* multiplexers are pre-configured with control values: 0010, 0111, 1101, and 1111.
- Outputs:** The circuit produces several outputs:
 - C1* (3 bits)
 - C1Z* (1 bit)
 - C2Z* (1 bit)
 - Z* (4 bits), the final sum output.

FIG. 203

705

DATA TYPE	BITS
Factory Code	16
Batch Number	32
Serial Number	48
Manufacturing Date	16
Media Length	24
Media Type	8
Preprinted Media Length	16
Cyan Ink Viscosity	8
Magenta Ink Viscosity	8
Yellow Ink Viscosity	8
Cyan Drop Volume	8
Magenta Drop Volume	8
Yellow Drop Volume	8
Cyan Ink Color	24
Magenta Ink Color	24
Yellow Ink Color	24
Remaining-media Length Indicator	16
Authentication Key	128
Copyrightable bit pattern	512
Reserved for Camera Use	88
TOTAL	1024

728

FIG. 204

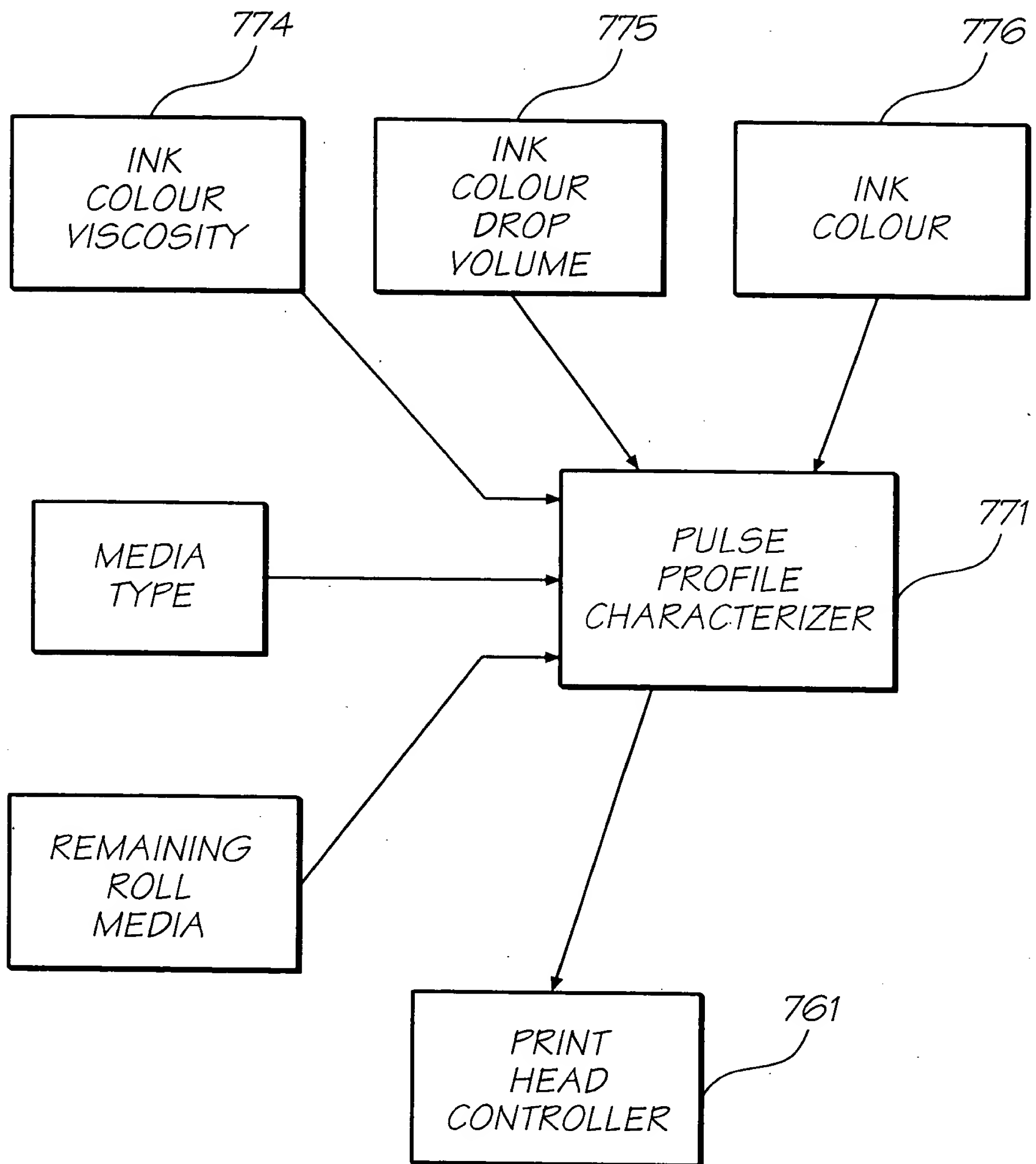


FIG. 205

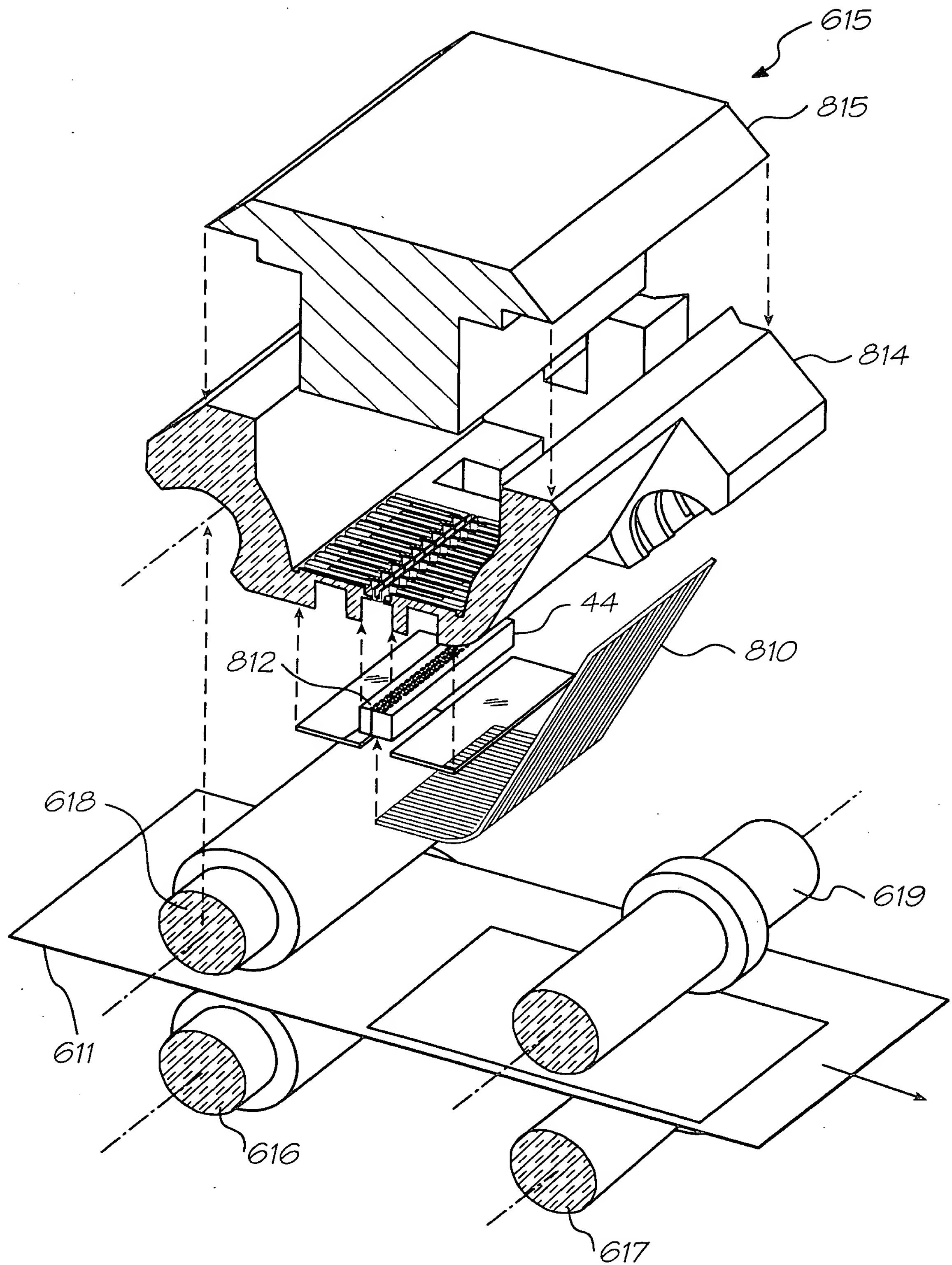


FIG. 206

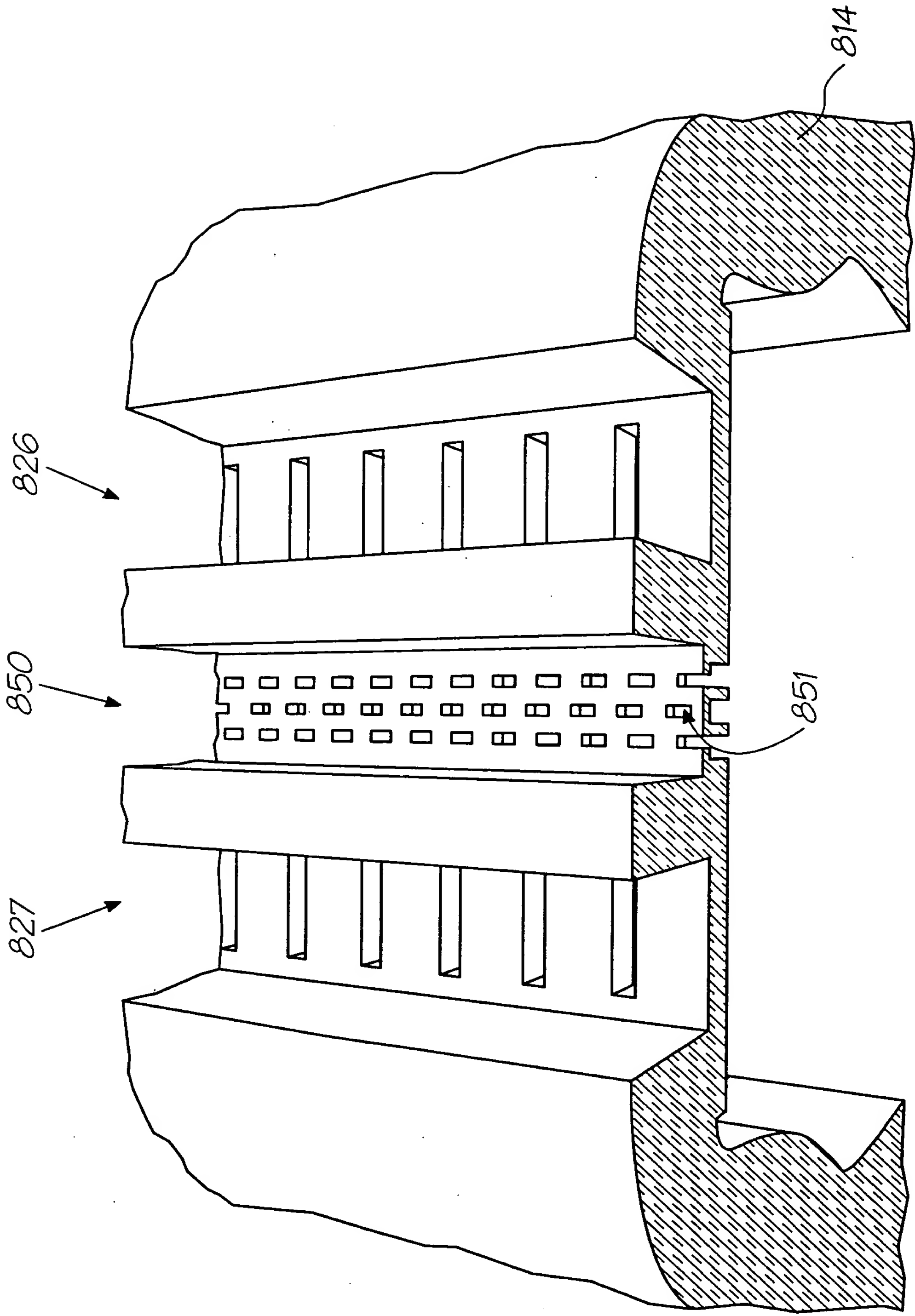


FIG. 208

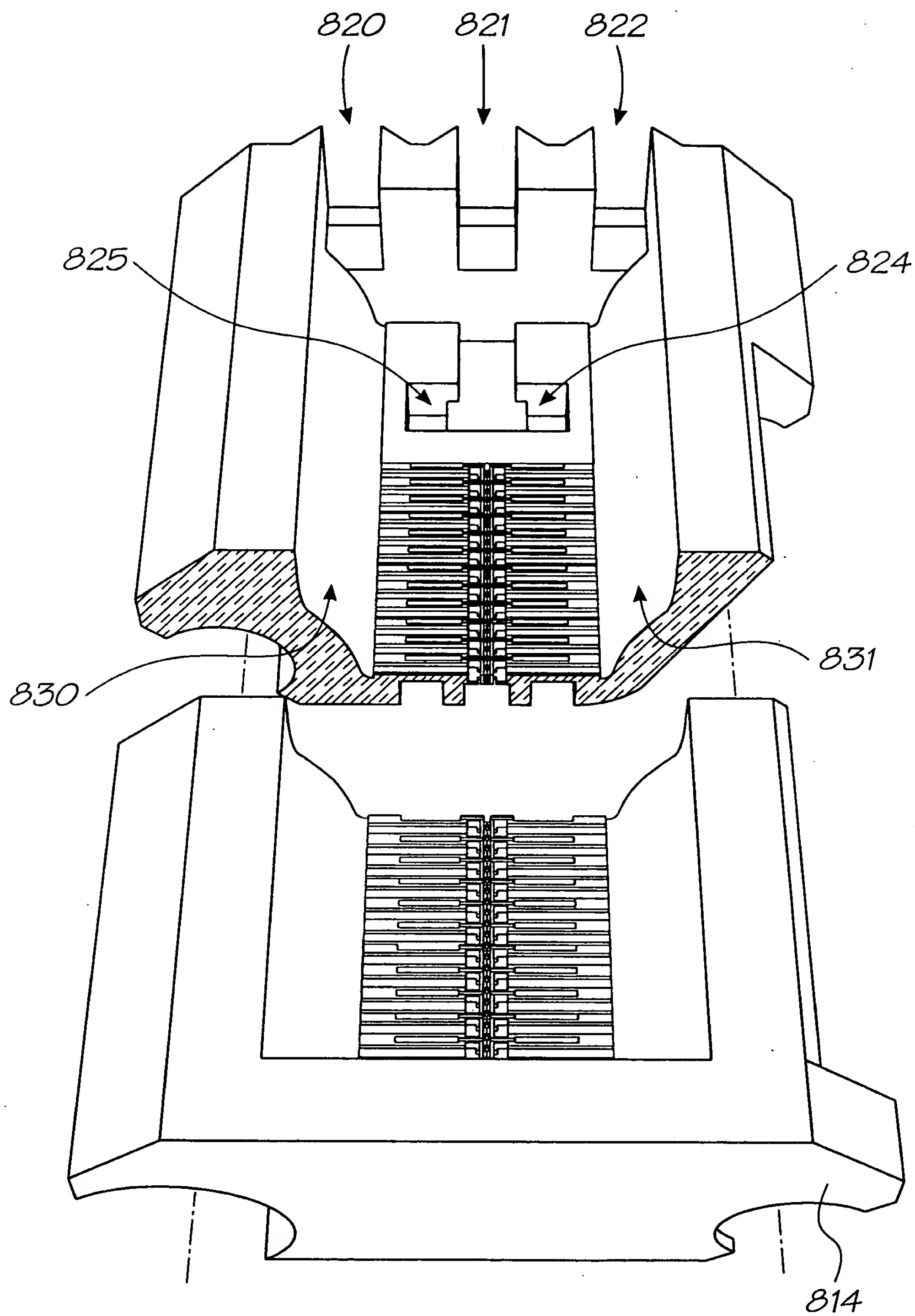


FIG. 209

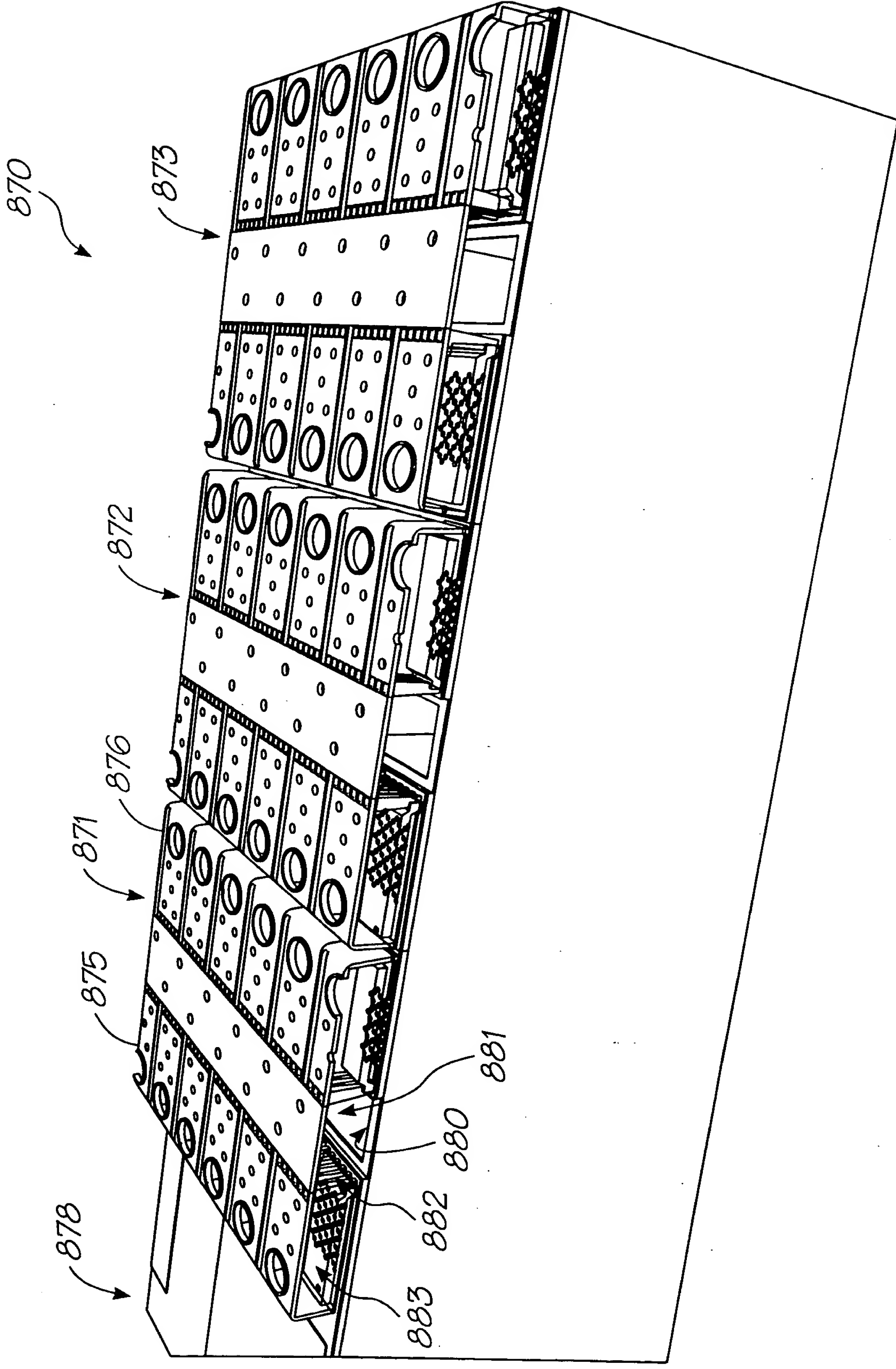


FIG. 211

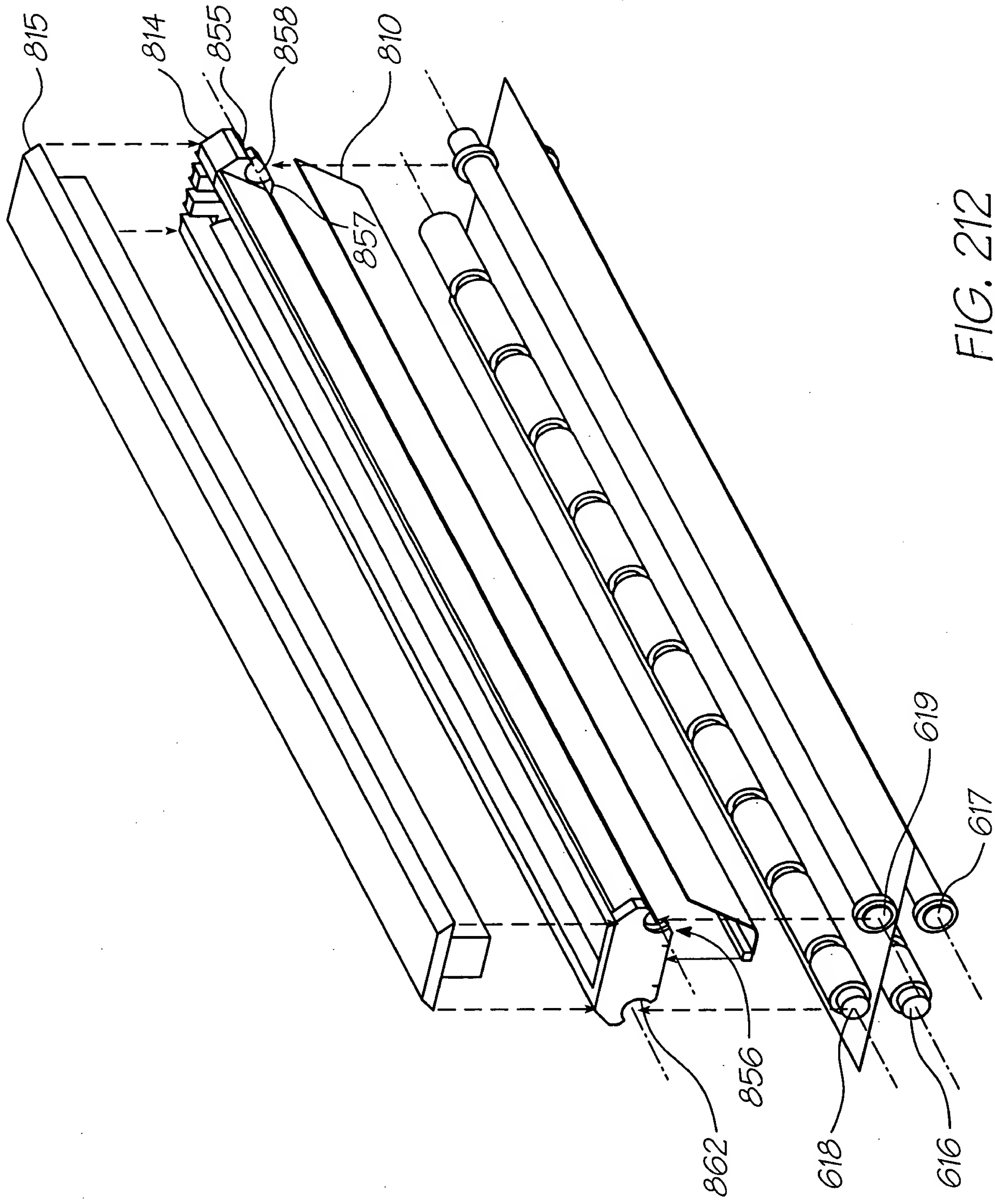


FIG. 212

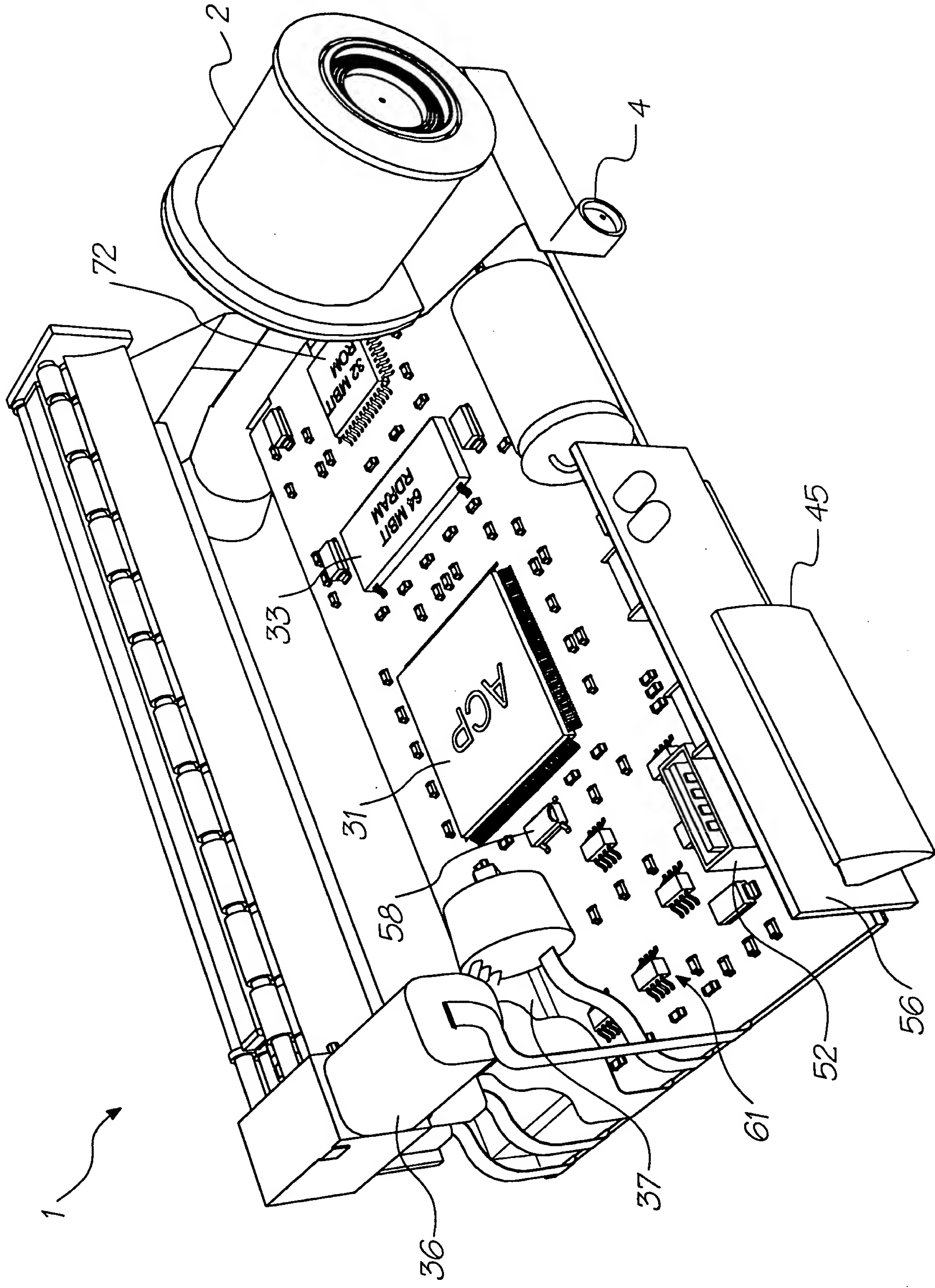


FIG. 215

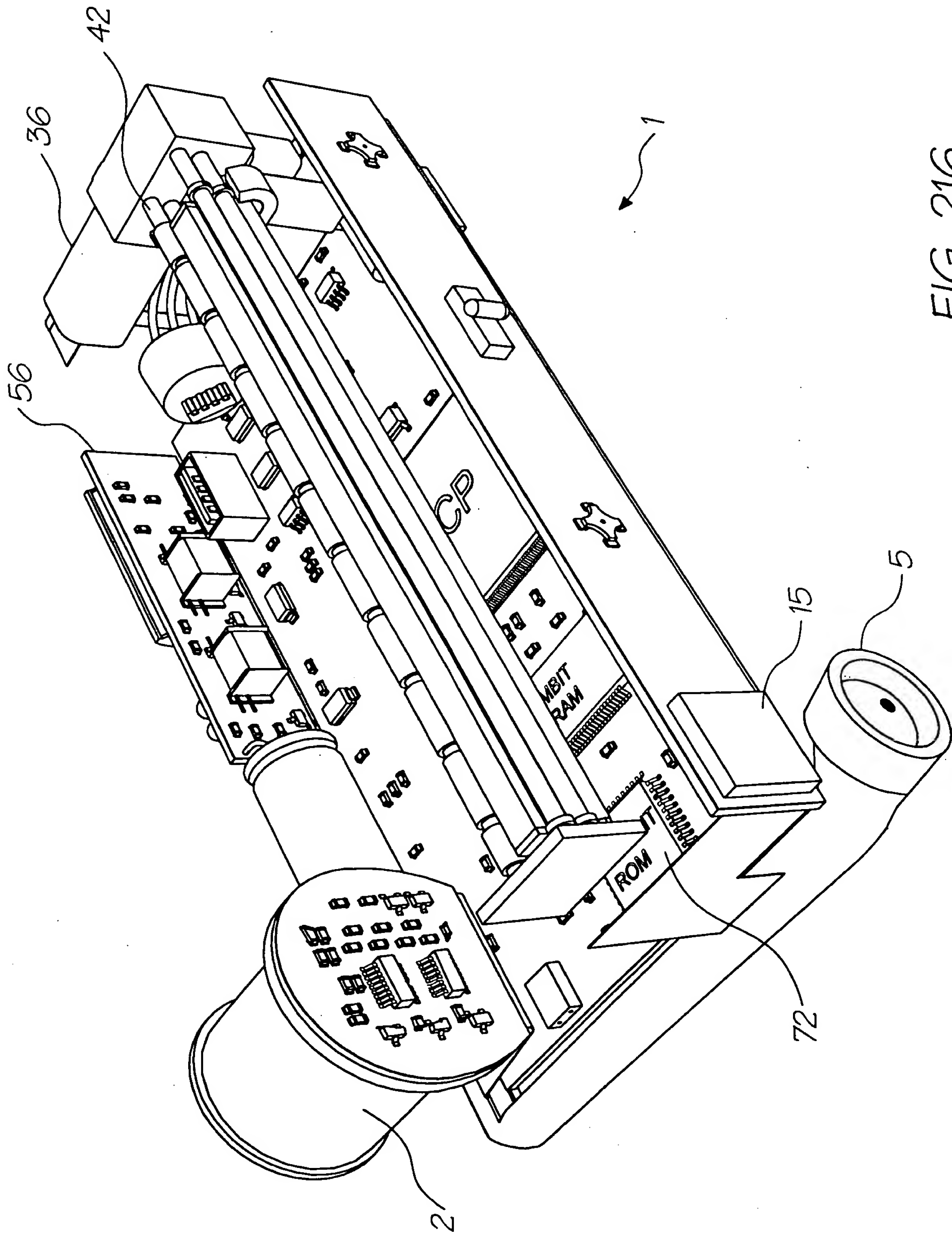


FIG. 216

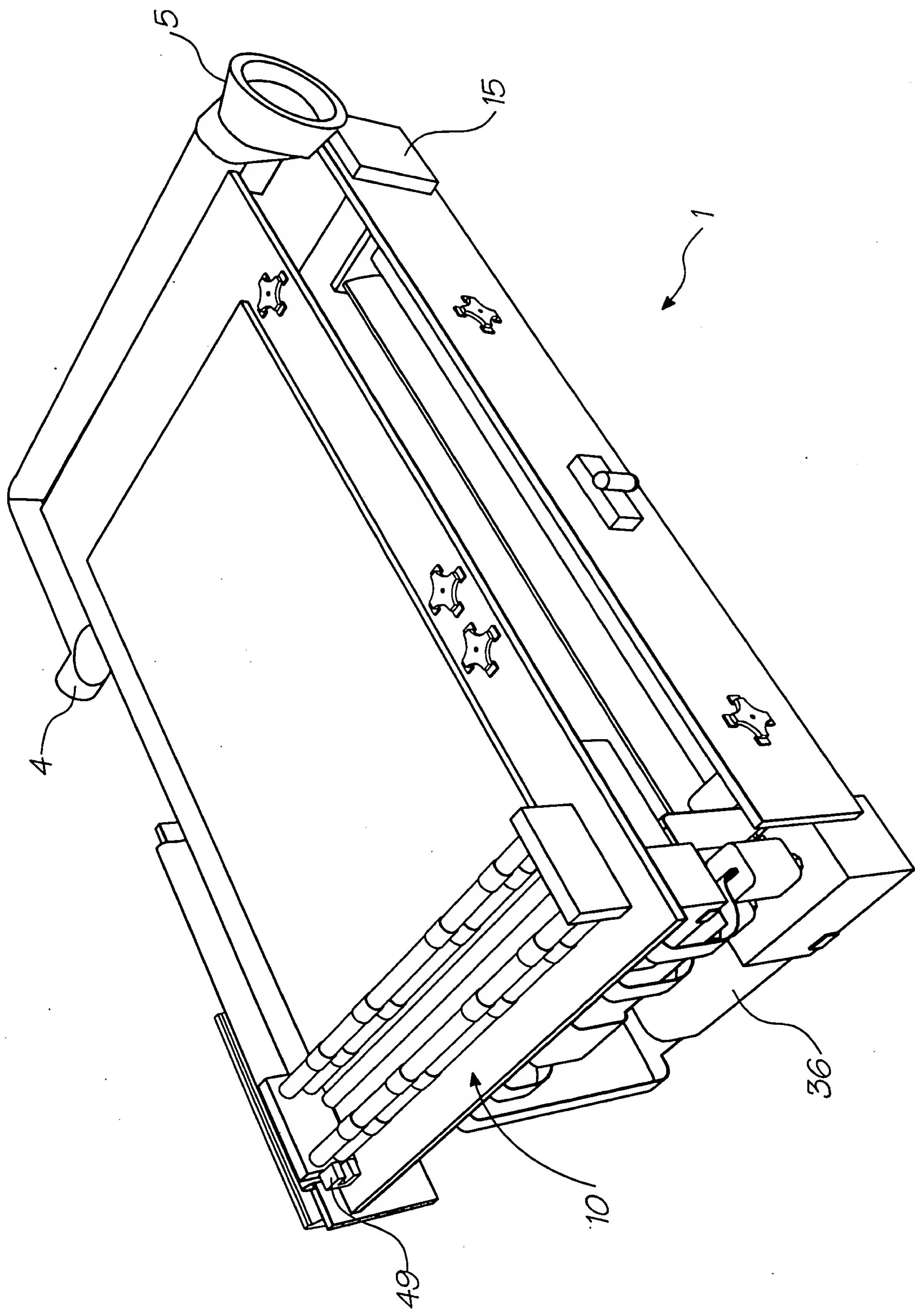


FIG. 217

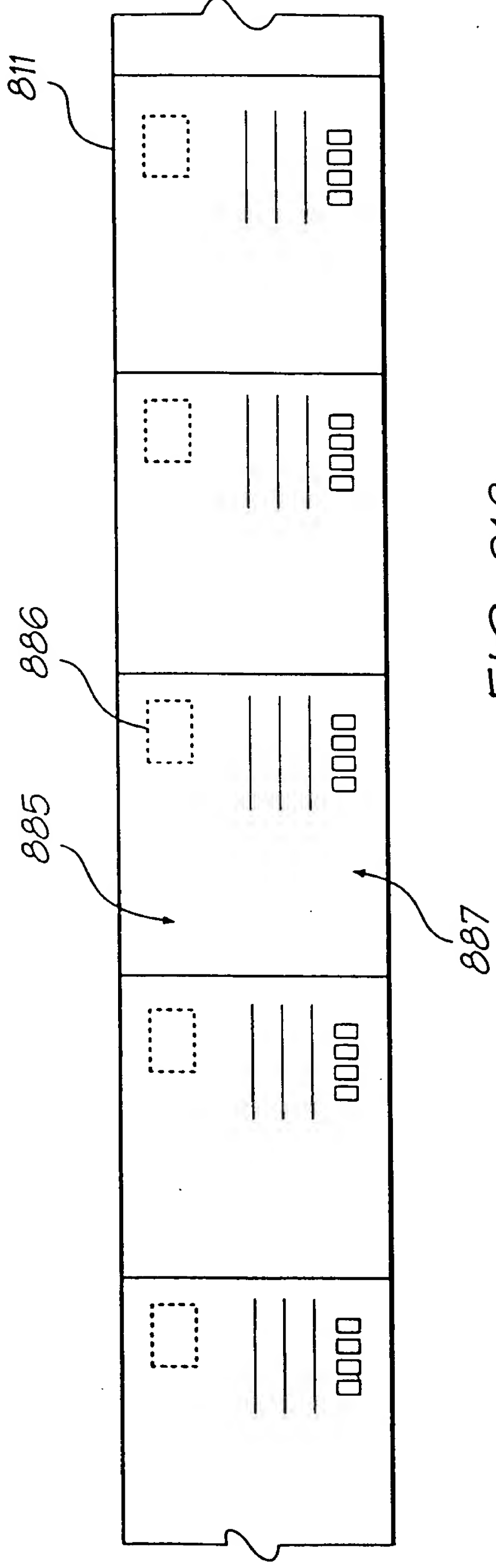


FIG. 218

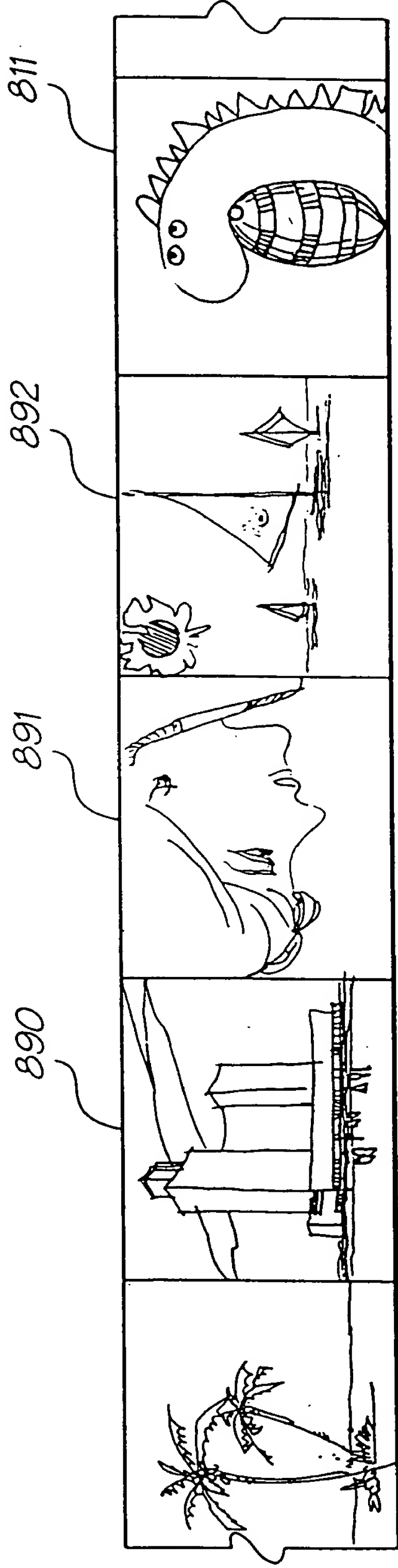


FIG. 219

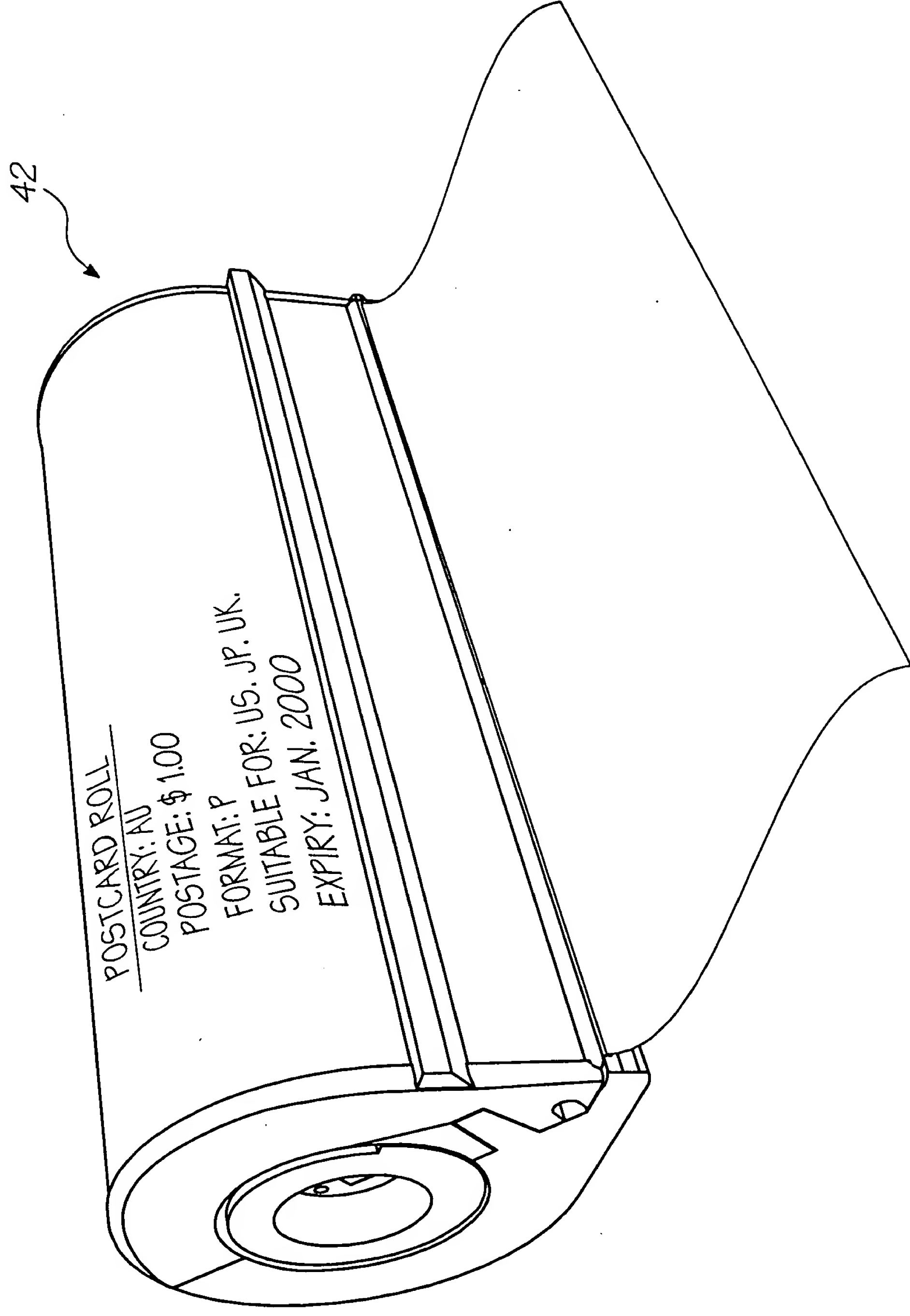


FIG. 220

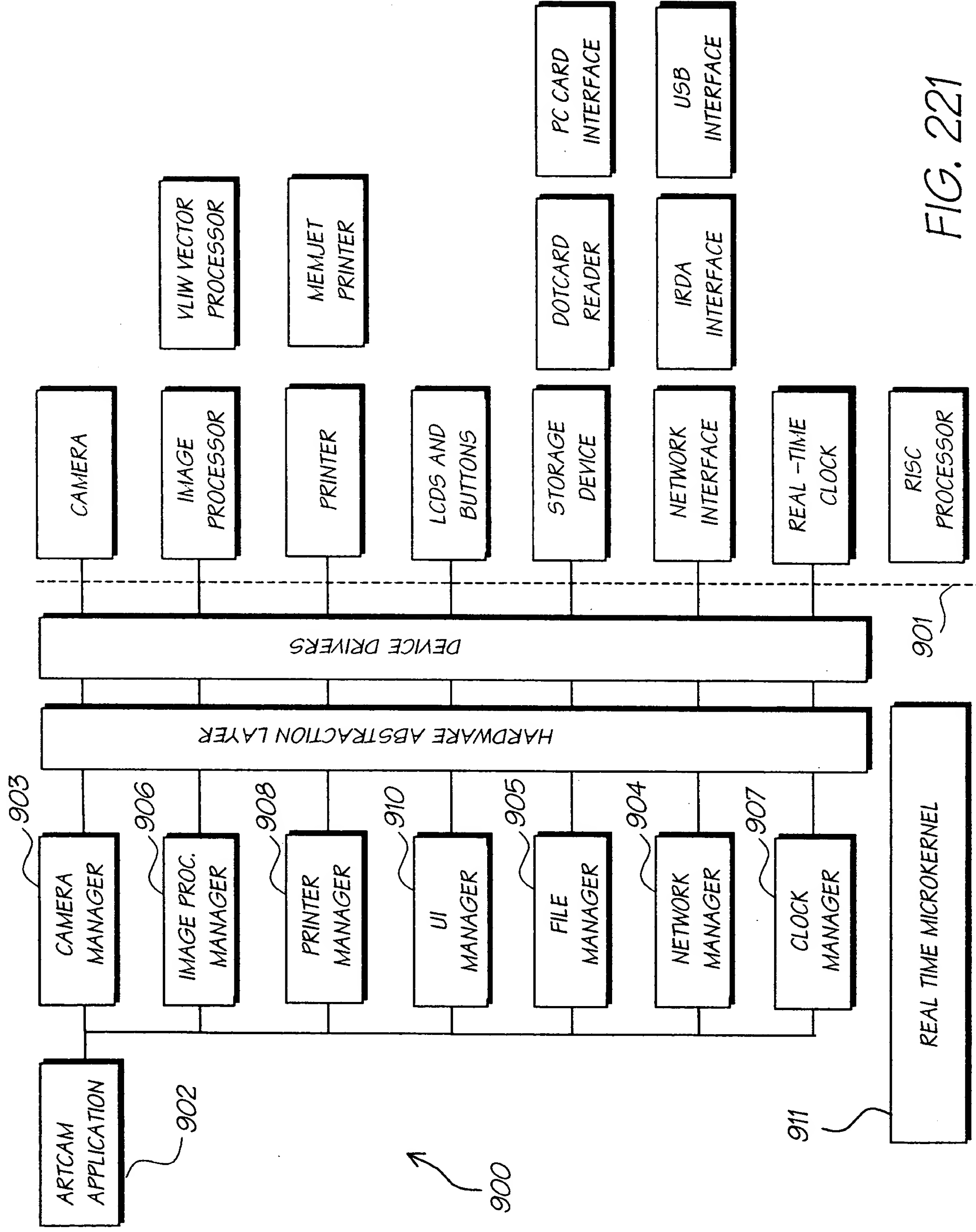


FIG. 221

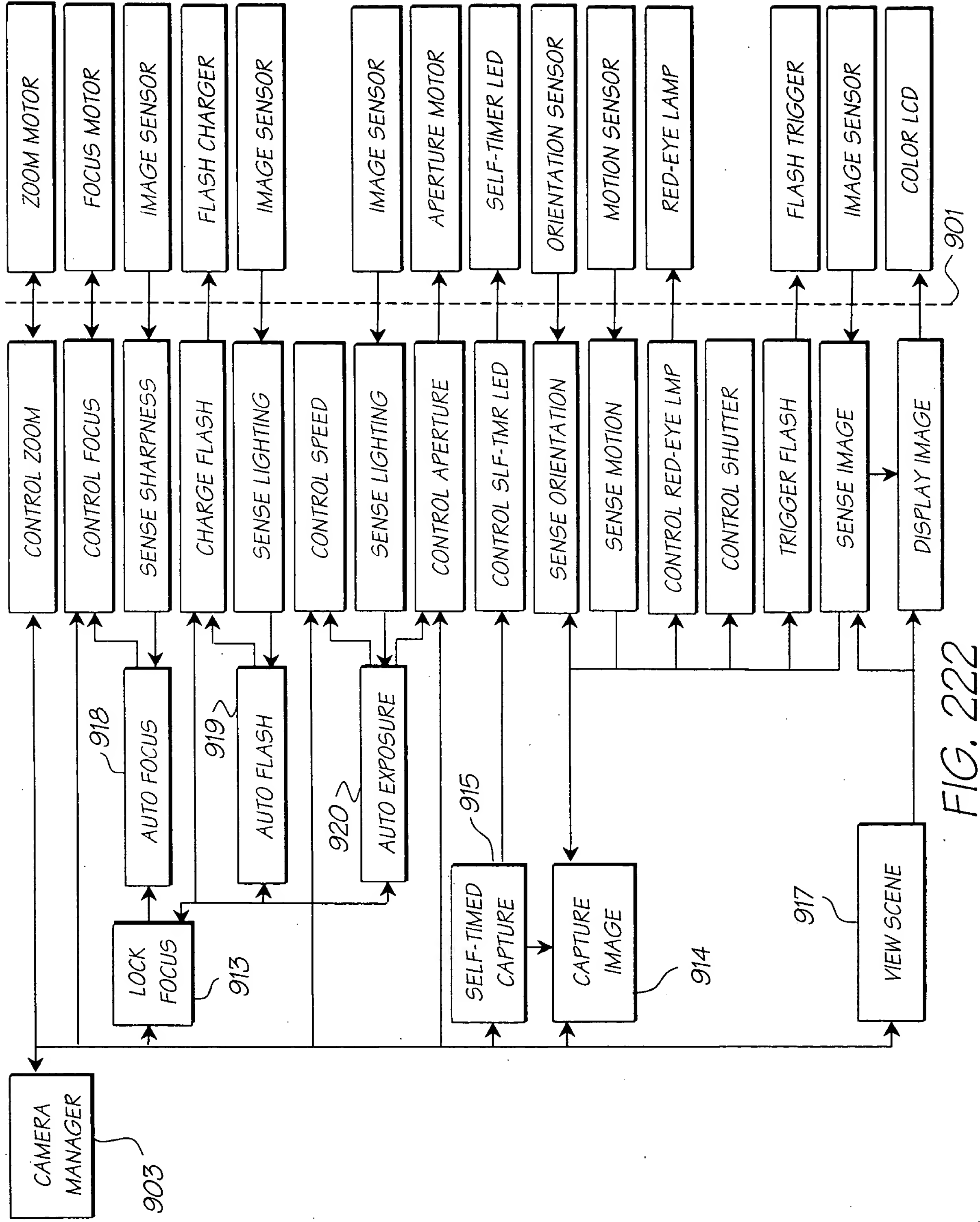


FIG. 222

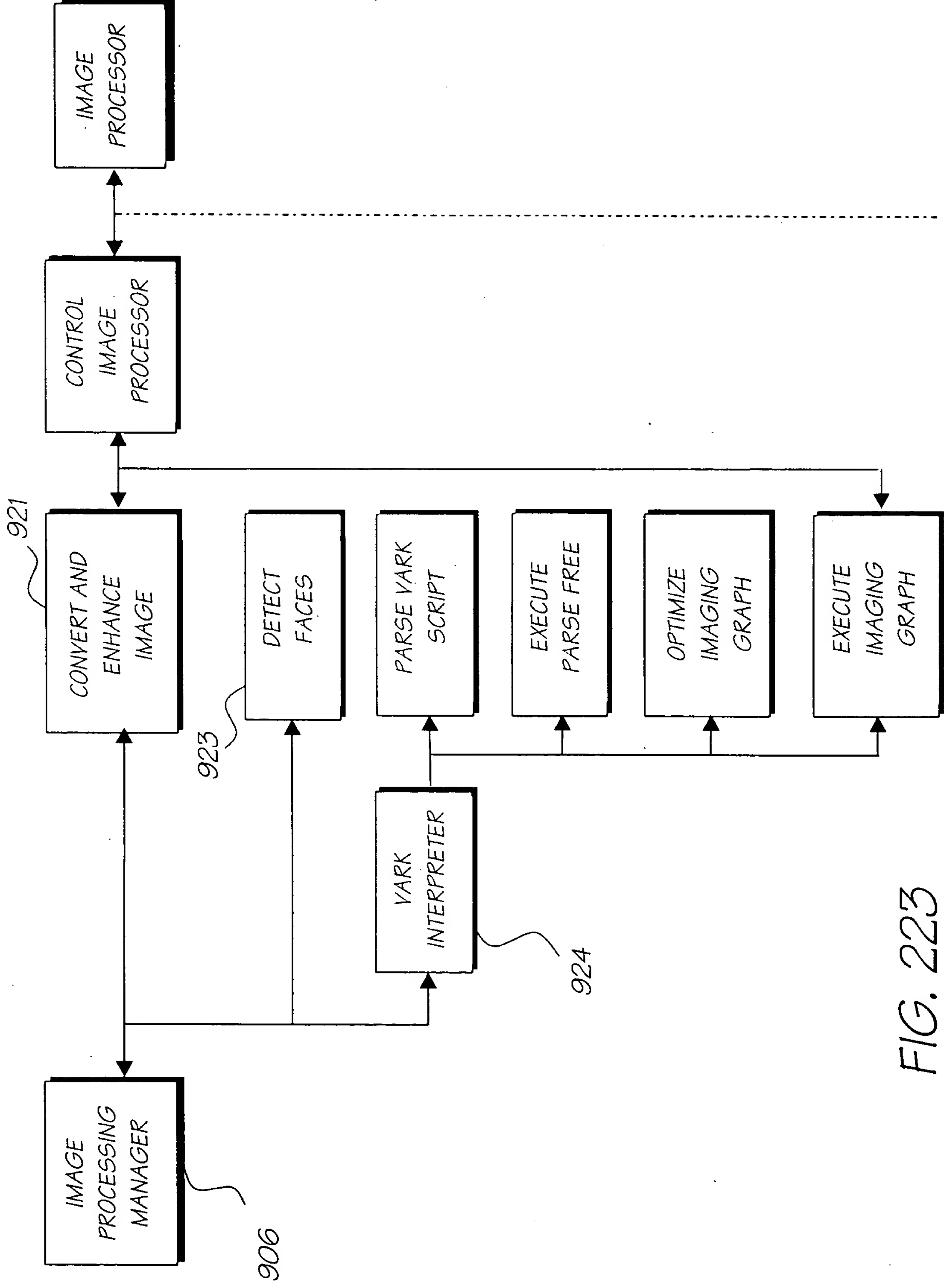


FIG. 223

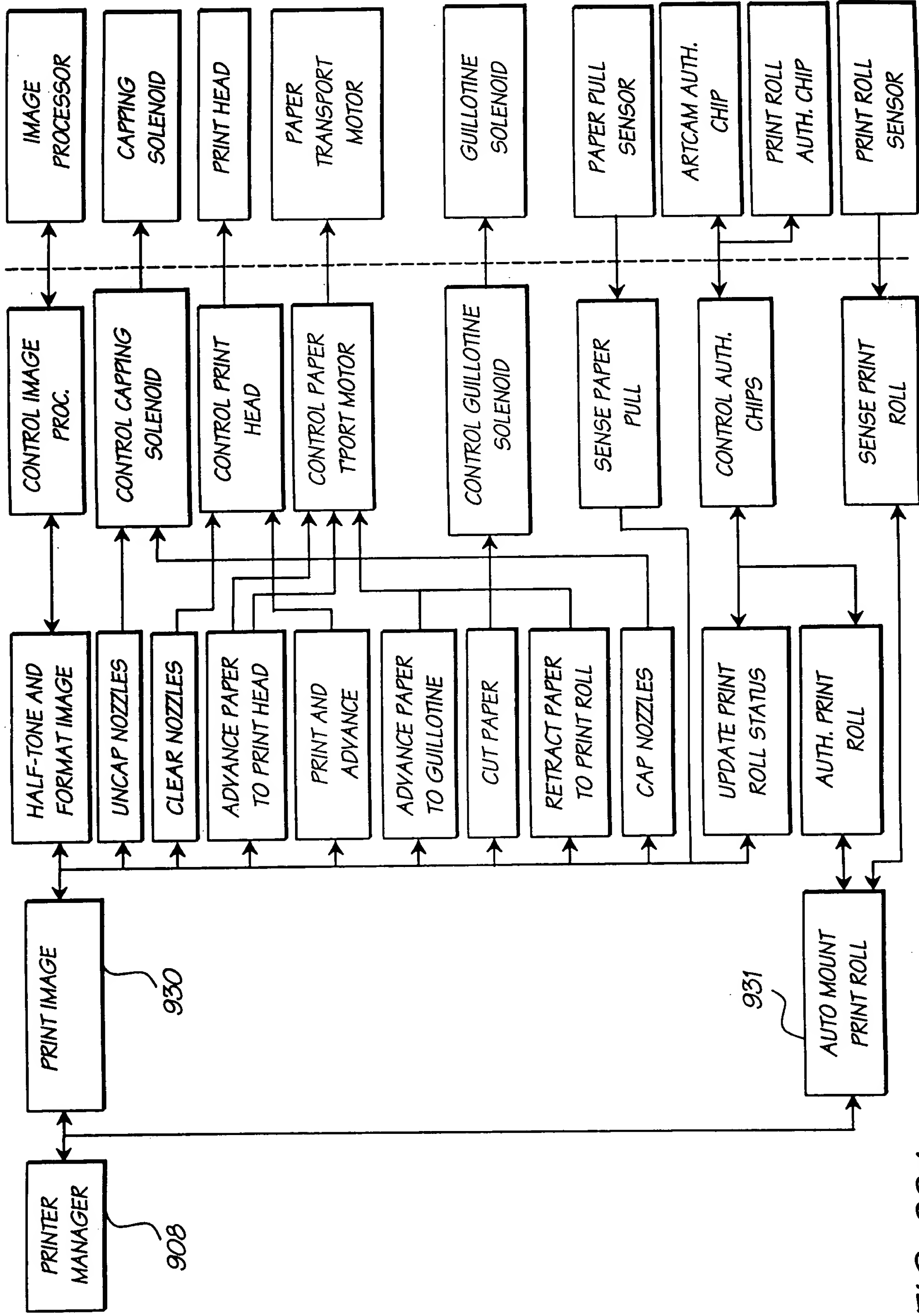


FIG. 224

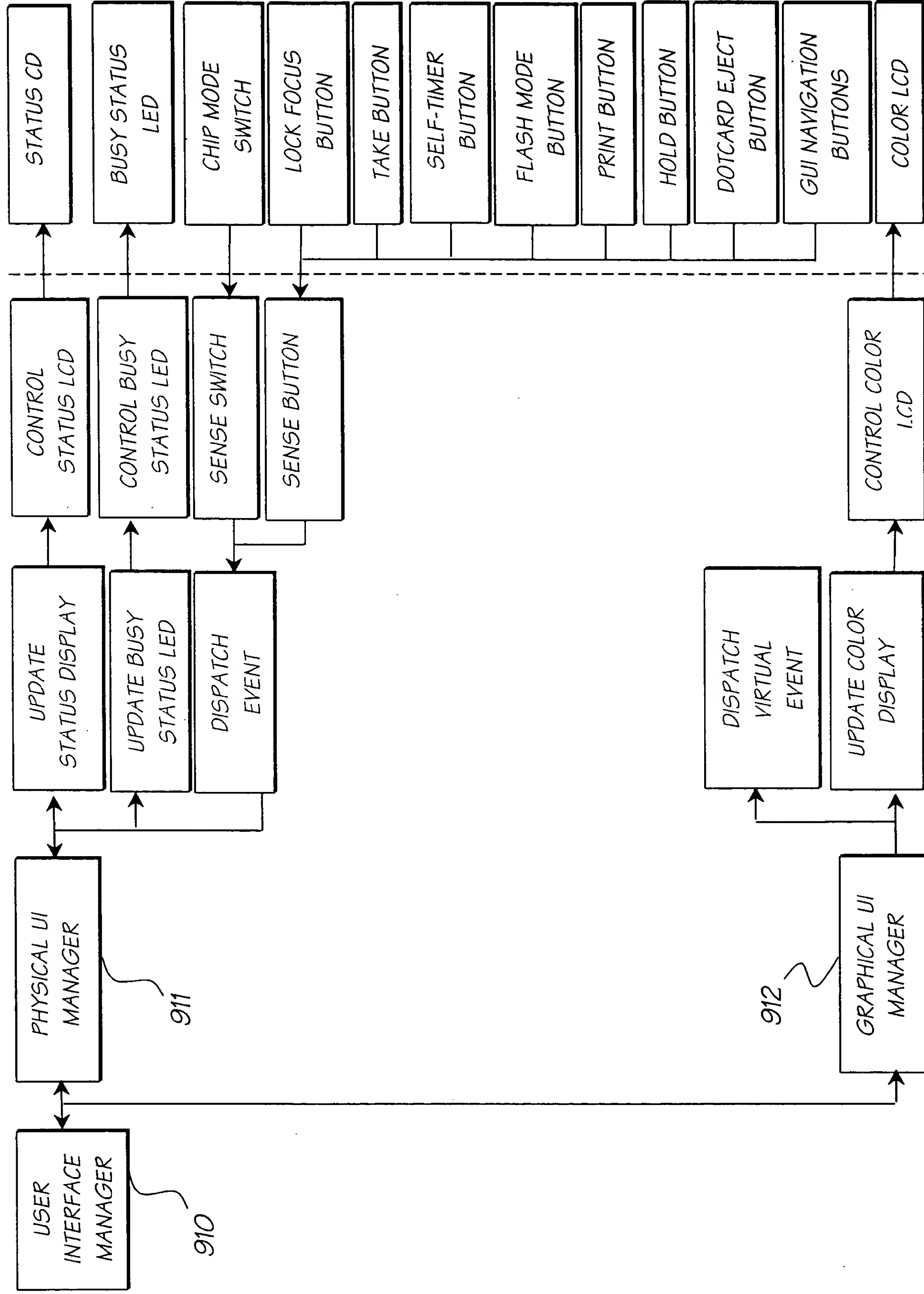


FIG. 225

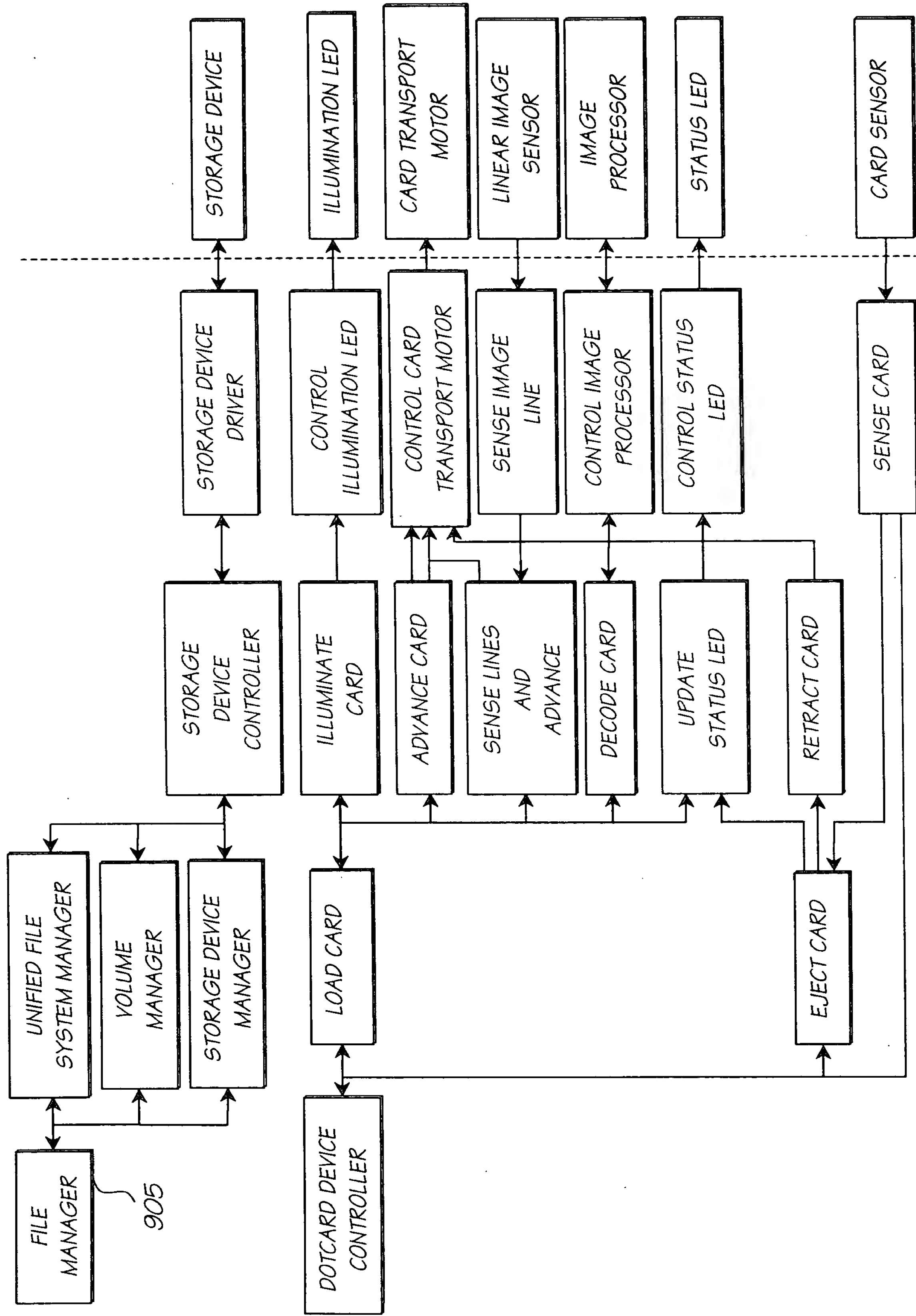


FIG. 226

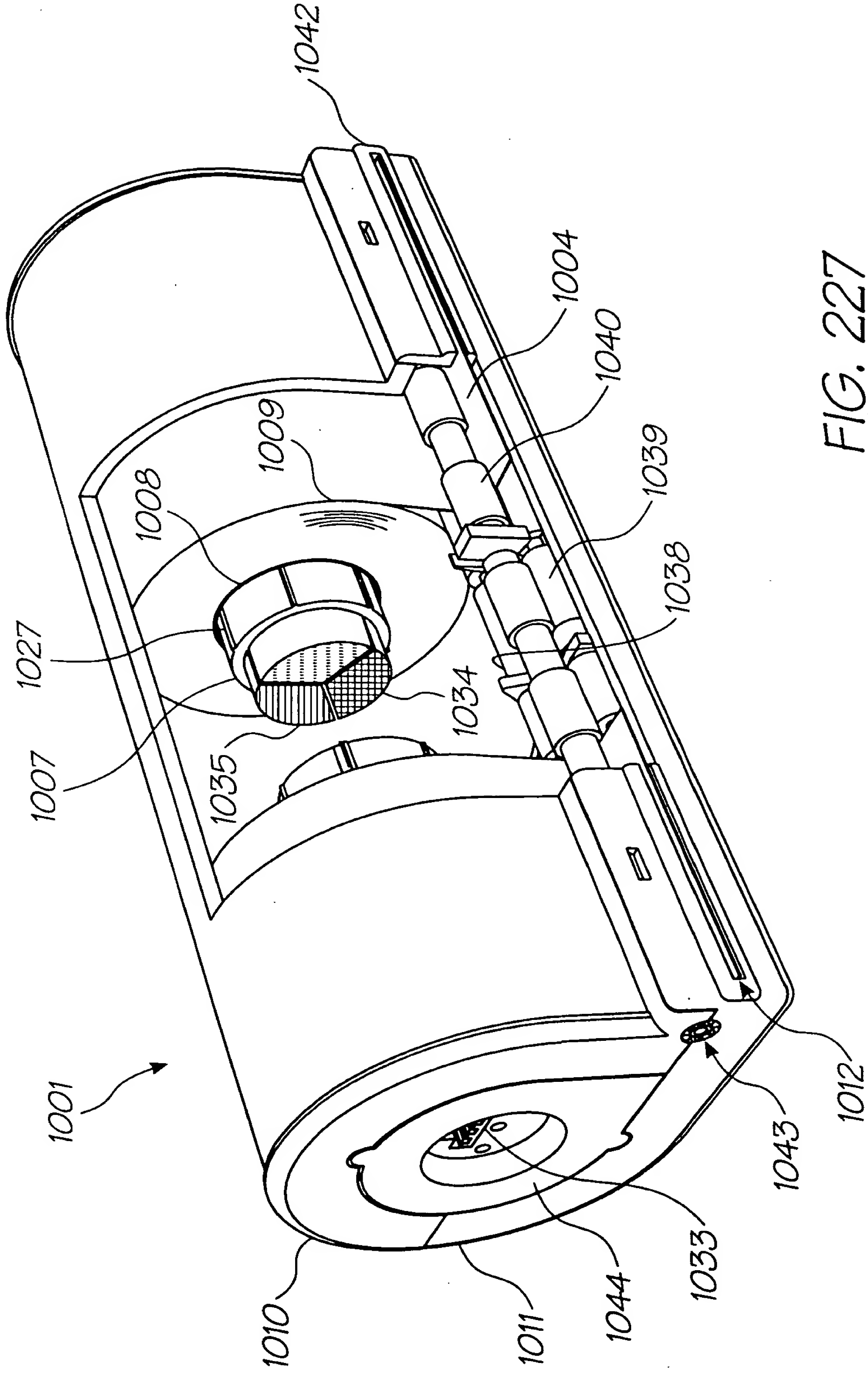


FIG. 227

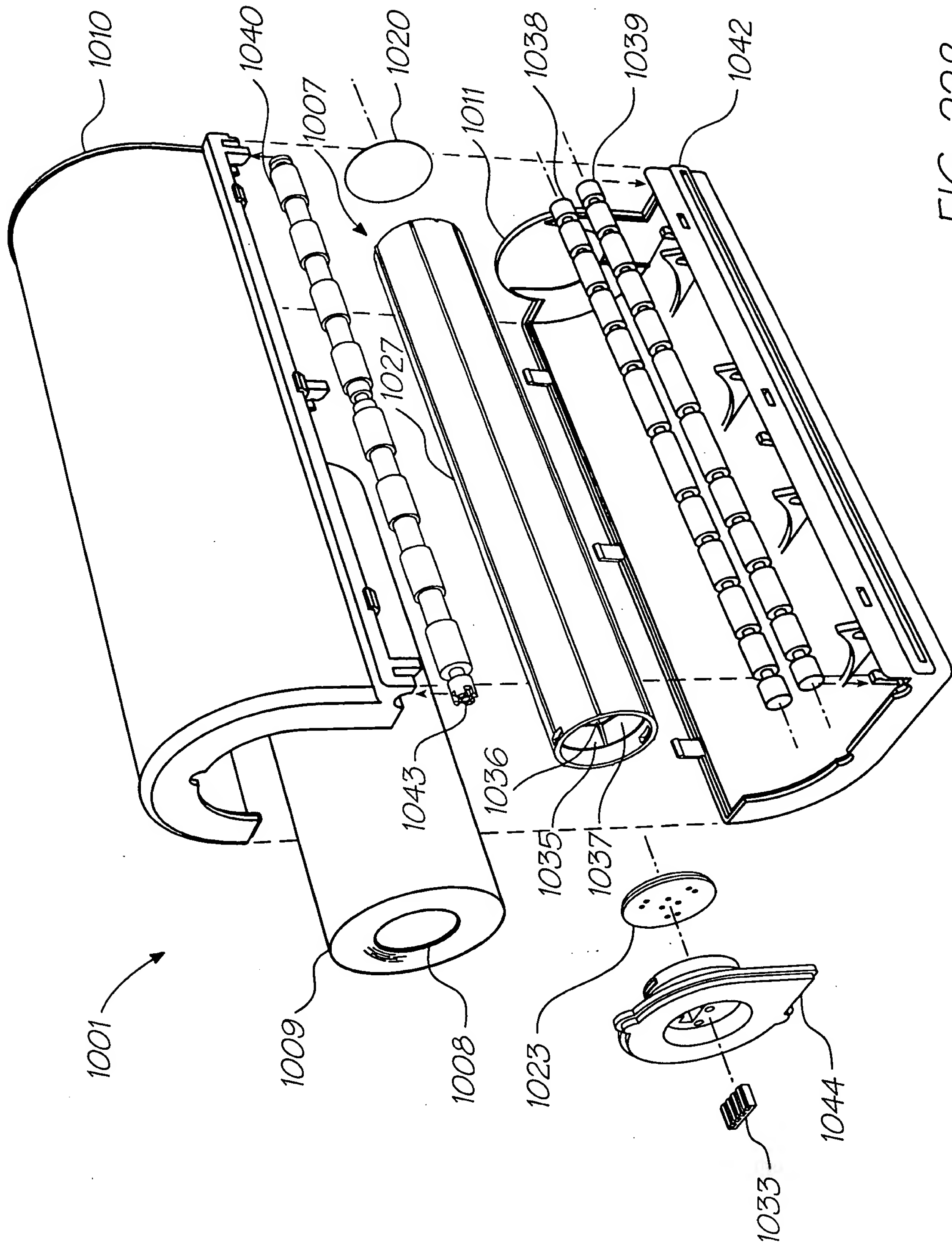


FIG. 228

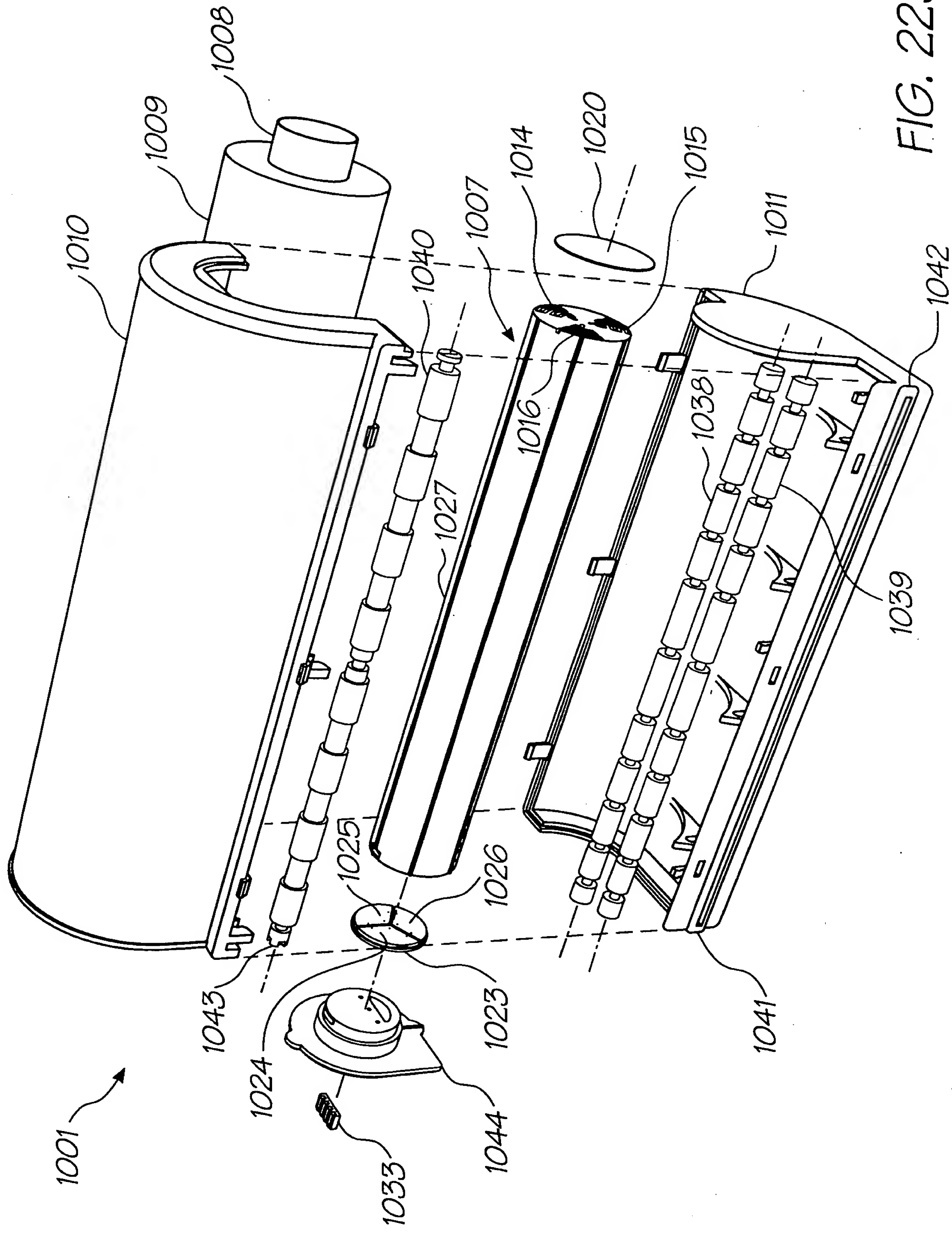


FIG. 229

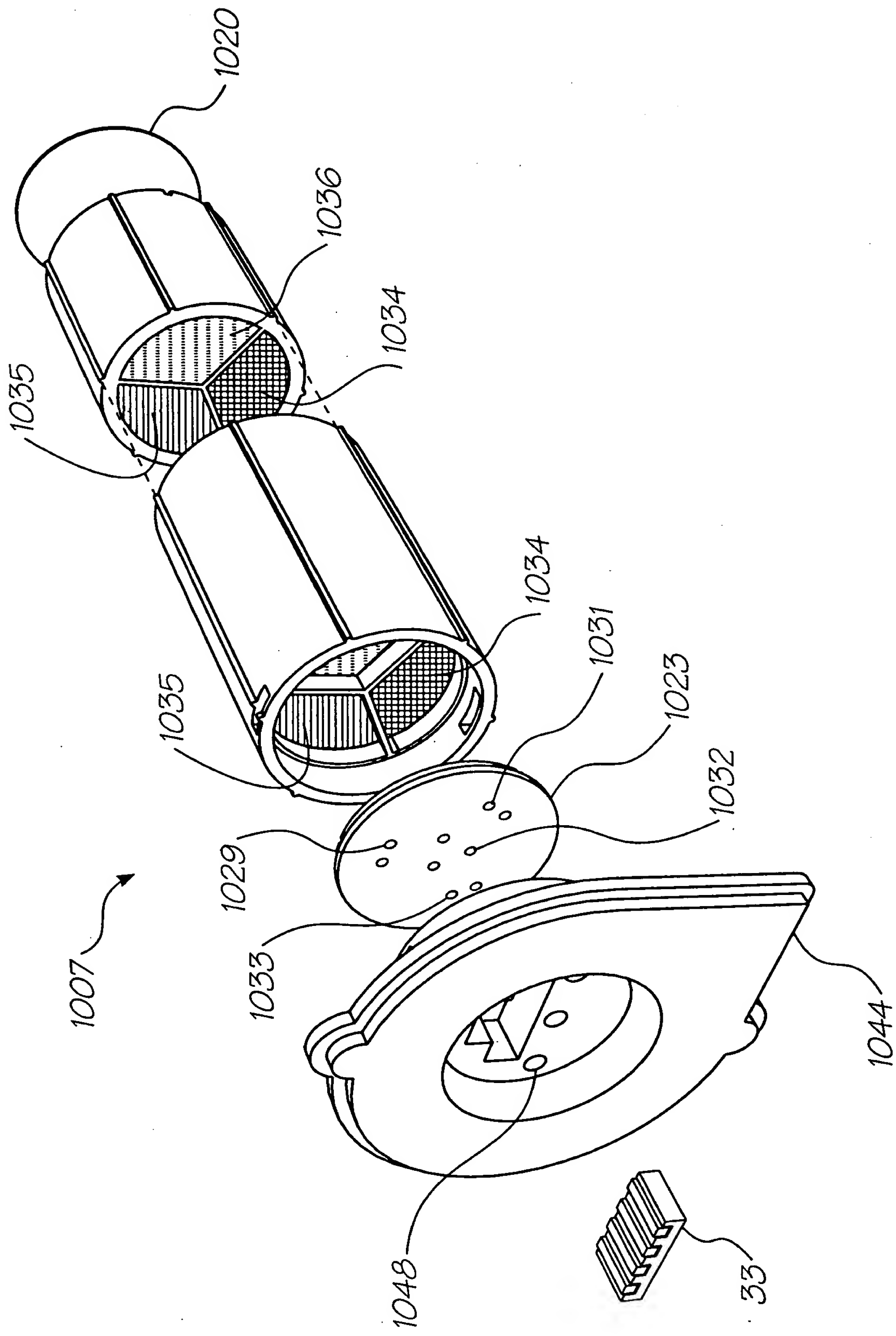


FIG. 230

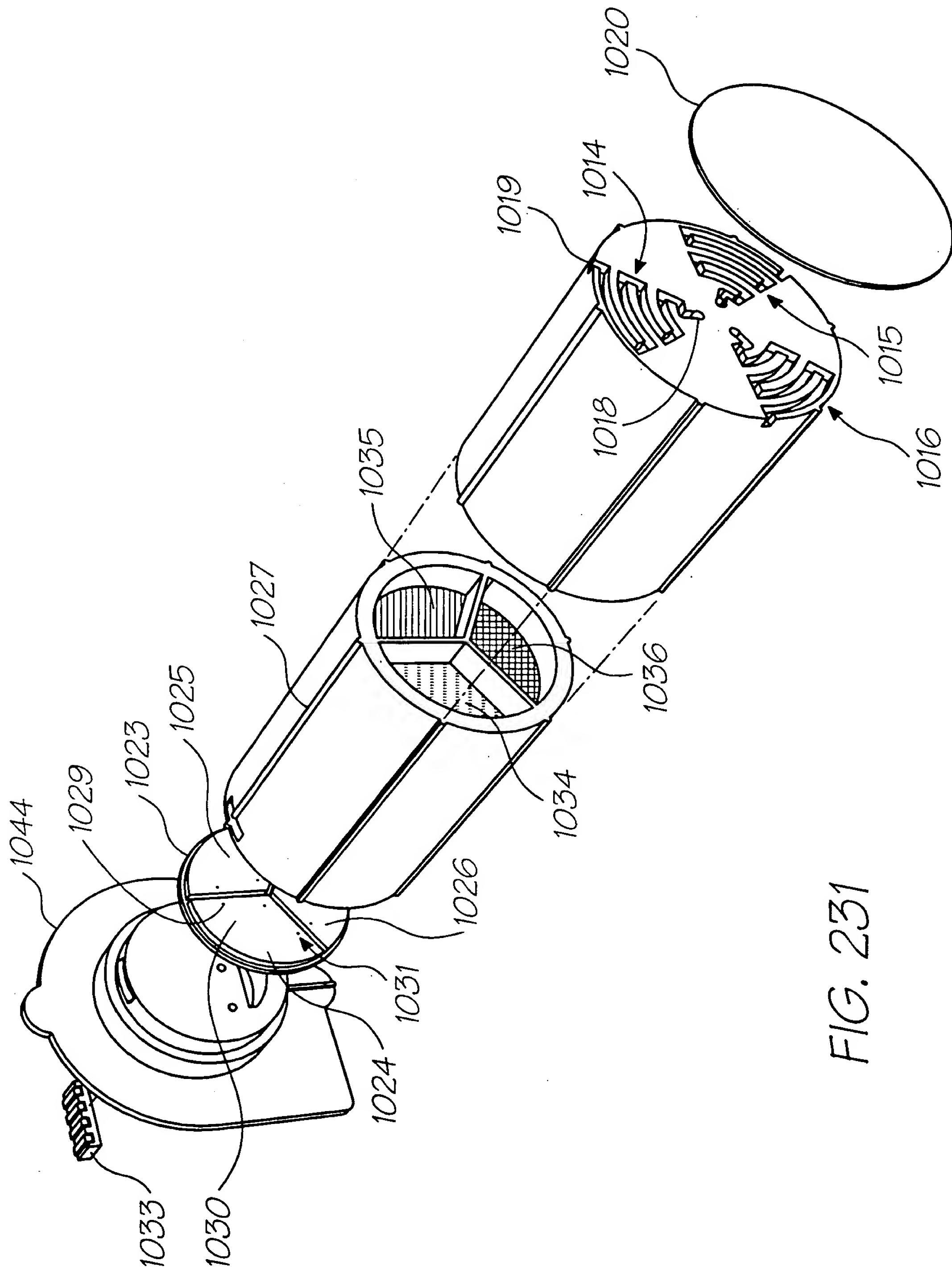


FIG. 231